

FACULTAT D'INFORMÀTICA DE BARCELONA

GRAU EN ENGINYERIA INFORMÀTICA

ESPECIALITAT EN ENGINYERIA DEL SOFTWARE

Implementació del pilot BIG IoT Barcelona: facilitant la interoperabilitat de l'IoT

Autor:
Àlex Català

Directora:
Dra. Mari Paz Linares
Ponent:
Dr. Ernest Teniente

26 de juny de 2018



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona

Agraïments

Agrair a l'inLab FIB l'oportunitat i confiança que m'ha permès realitzar aquest treball final de grau dins d'un projecte tan engrescador.

Agrair als companys i família que m'han acompanyat durant la carrera i aquest treball final.

També agrair a l'Ernest els seus consells i ocurrencies, sempre interessants.

I, finalment, agrair en especial a la Mari Paz, en Juan i en Xavi, un equip genial que m'ha permès créixer i gaudir d'aquests dos últims anys.

Resum

L'Internet de les coses permet la interconnexió entre dispositius i persones, generant així informació valuosa que és utilitzada per millorar i automatitzar tot tipus de tasques de la nostra vida quotidiana i professional. Aquests beneficis han causat un augment de la presència de la tecnologia en àmbits i disciplines fins ara poc o gens informatitzats; incrementant, doncs, el nombre de dispositius interconnectats. Aquest creixement presenta problemàtiques diverses que requereixen un canvi en la manera d'intercanviar les dades.

BIG IoT és un projecte europeu que investiga les possibles solucions als reptes que presenten aquestes problemàtiques, posant especial atenció en facilitar la interoperabilitat entre plataformes.

L'inLab FIB participa a BIG IoT desenvolupant el pilot de Barcelona en que s'analitzen i estudien les solucions proposades pel projecte europeu. En aquest treball final de grau s'implementen les aplicacions i serveis que constitueixen part del pilot de Barcelona, fent ús de les eines i ecosistema proporcionats per BIG IoT.

Resumen

El Internet de las cosas permite la interconexión entre dispositivos y personas, generando así información valiosa utilizada para mejorar y automatizar todo tipo de tareas. Estos beneficios han causado un aumento de la presencia de la tecnología en áreas y disciplinas hasta ahora poco o nada informatizadas; incrementando, así, el número de dispositivos interconectados. Este crecimiento presenta problemas diversos que requieren un cambio en la manera de intercambiar los datos.

BIG IoT es un proyecto europeo que investiga las posibles soluciones a los retos que presentan los problemas mencionados, poniendo especial atención en facilitar la interoperabilidad entre plataformas.

InLab FIB participa a BIG IoT desarrollando el piloto de Barcelona, en que se analizan y estudian las soluciones propuestas por el proyecto europeo. En este trabajo final de grado se implementan las aplicaciones y servicios que constituyen parte del piloto de Barcelona, utilizando las herramientas y ecosistema proporcionados por BIG IoT.

Abstract

The Internet of Things allows the interconnection between devices and humans. This network provides valuable information that is used to automate and improve the quality of all kind of tasks. These benefits are bringing technology into new fields, increasing the number of connected devices. This device's growth is making more difficult to process and integrate the data provided by different sources.

The european project BIG IoT aims to find solutions on the issues presented by the mentioned growth. In fact, the BIG IoT main focus is to solve the interoperability issue between platforms.

inLab FIB participates in BIG IoT developing the Barcelona pilot. This proof of concept wants to use and analyse the solutions provided by BIG IoT. This final degree project will design and implement the software components that are necessary to complete the BIG IoT Barcelona pilot.

Índex

1	Introducció	5
2	Context: BIG IoT	6
3	Estat de l'art	8
4	Motivació i objectius	10
4.1	Motivació	10
4.2	Objectius	11
5	Abast	12
5.1	Possibles obstacles	15
6	<i>Stakeholders</i>	16
7	Metodologia i rigor	17
7.1	Metodologies de desenvolupament	17
7.2	Control de gestió	18
7.3	Valoració d'alternatives i pla d'acció	18
7.4	Eines de seguiment	19
8	Especificació	20
8.1	Objectius de les parts interessades	20
8.2	Requisits no funcionals	22
8.3	Requisits funcionals	23
8.3.1	<i>Traffic Information Center Tool</i>	23
8.3.2	<i>BCN Traffic Info</i>	23
8.4	Funcionalitats	24
8.4.1	Serveis BIG IoT	24
8.4.2	<i>Traffic Information Center Tool</i>	26
8.4.3	<i>BCN Traffic Info</i>	29
8.5	Model conceptual	31
9	Disseny	34
9.1	Arquitectura	34
9.1.1	Servei d'alarmes	35
9.2	Tecnologia	37
9.3	Patrons	39
9.4	Diagrames de classes	41
9.4.1	<i>Traffic Information Center Tool</i>	41

9.4.2	<i>BCN Traffic Info</i>	44
10	Implementació	47
10.1	Tasques a realitzar	47
10.2	Exemples de codi BIG IoT	50
11	Desplegament	54
11.1	Servidor de producció i eines de monitoratge	54
11.2	Integració contínua	56
12	Validacions	59
12.1	Validacions de les parts interessades	59
12.2	<i>Testing</i>	59
13	Planificació temporal i econòmica	62
13.1	Planificació general	62
13.2	Recursos	62
13.2.1	Personal	62
13.2.2	Material	63
13.3	Planificació econòmica	66
13.3.1	Recursos de personal	66
13.3.2	Costos indirectes	67
13.3.3	Imprevistos	67
13.3.4	Contingències	68
13.3.5	Pressupost final	69
14	Anàlisi de la sostenibilitat	70
14.1	Econòmica	70
14.2	Ambiental	70
14.3	Social	71
15	Conclusions	72
15.1	Assoliment d'objectius	72
15.2	Contribucions	73
15.3	Treball futur	73
15.4	Valoració personal	74
A	Captures de pantalla del <i>Traffic Information Center Tool</i>	77
B	Captures de pantalla de la <i>BCN Traffic Info</i>	82

Índex de figures

1	Vista general dels components a desenvolupar	12
2	Diagrama de casos d'ús <i>Traffic Information Center Tool</i>	27
3	Diagrama de casos d'ús <i>BCN Traffic Info.</i>	30
4	Model conceptual	33
5	Arquitectura general	35
6	Arquitectura del servei d'alarmes	37
7	Diagrama de classes del <i>backend</i> del <i>Traffic Information Center Tool</i>	42
8	Diagrama de classes del <i>frontend</i> del <i>Traffic Information Center Tool</i>	43
9	Diagrama de classes del <i>backend</i> de la <i>BCN Traffic Info</i>	45
10	Diagrama de classes del <i>frontend</i> de la <i>BCN Traffic Info</i>	46
11	Panell de monitoratge Graphana	55
12	Panell de gestió Docker Portainer	56
13	Configuració Gitlab-CI <i>Traffic Information Center Tool</i>	57
14	Diagrama de Gantt 1	64
15	Diagrama de Gantt 2	65
16	Pàgina de benvinguda	77
17	Vista del mapa amb dades de mobilitat 1	78
18	Vista del mapa amb dades de mobilitat 2	78
19	Vista informació de l'oferta de dades	79
20	Vista del mapa amb dades de contaminació	79
21	Vista gestió recomanacions	80
22	Vista creació alarmes (opció aparcament)	80
23	Vista creació alarmes (recomanació associada)	81
24	Vista gestió alarmes	81
25	Vista ruta i temps de viatge amb recomanacions actives	82
26	Vista recomanacions en ruta	83
27	Vista de totes les recomanacions actives a la ciutat	84

Índex de taules

1	Taula indicadors integració BIG IoT	61
2	Taula de recursos materials	63
3	Recursos personal	66
4	Recursos hardware	67
5	Recursos indirectes	67
6	Taula contingències	68
7	Taula pressupost total	69

Glossari

- **Framework:** Entorn de treball que facilita el desenvolupament del software d'un àmbit o problemàtica concreta.
- **Frontend:** Concepte que fa referència a la separació entre la capa de presentació i la d'accés a les dades. Concretament *frontend* fa referència a la capa de presentació.
- **Backend:** Concepte que fa referència a la separació entre la capa de presentació i la d'accés a les dades. Concretament *backend* fa referència a la capa d'accés a les dades.
- **API:** Acrònim d'Application Programming Interface, que fa referència a funcionalitats i mètodes que ofereixen una llibreria.

1 Introducció

Durant els últims anys el nombre de dispositius connectats a Internet ha crescut enormement, ja són moltes les prediccions que indiquen el creixement exponencial del nombre d'aquests dispositius i, en conseqüència, el mercat de l'*Internet Of Things (IoT)*[16].

Aquests dispositius ens proporcionen dades de diversos camps; *Smart Cities*, *Connected Health*, *Industrial IoT*, *Connected Cars*, *Wearables* i *Smart Homes* entre molts altres. Aquestes dades han esdevingut molt útils per a la presa de decisions a empreses, administracions i usuaris. Ens proporcionen informació valuosa que ajuda a millorar els models de negoci i la nostra vida diària.

Actualment cada empresa ofereix les seves dades de manera independent, sense seguir cap estàndard o patró de comunicació que faciliti la interoperabilitat entre plataformes. És a dir, les comunicacions entre proveïdors i consumidors de dades és vertical, dificultant així processos d'automatització com la cerca i/o integració dels diferents proveïdors de dades. Aquest escenari, amb l'oferta de dades totalment descentralitzada, complica el desenvolupament de plataformes que siguin capaces d'explotar el gran potencial que presenta la informació provinent de l'encreuament entre dades de diferent tipologia o origen.

El projecte europeu BIG IoT[2] (Bridging the Interoperability Gap of the Internet of Things) vol donar resposta a la problemàtica plantejada mitjançant una solució que busca centralitzar el mercat i facilitar la interoperabilitat entre les diferents plataformes que hi intervenen.

A BIG IoT es desenvolupen tres proves pilot amb la finalitat de demostrar la viabilitat del projecte; la regió de Piedmont (Itàlia), la zona nord d'Alemanya i la ciutat de Barcelona.

En aquest treball final de grau s'implementa part del pilot de Barcelona, i mitjançant casos d'ús relacionats amb la mobilitat i les *Smart Cities*, es vol demostrar que el projecte BIG IoT seria viable i útil per a la ciutat.

2 Context: BIG IoT

El projecte europeu BIG IoT [2] és part de la IoT-EPI [8], iniciativa europea per incentivar el desenvolupament de plataformes IoT. Aquest programa està format per diversos projectes que tenen com a objectiu millorar l'ecosistema IoT europeu facilitant-ne el desenvolupament, la interoperabilitat entre plataformes i la manera en que es distribueix la informació. Aquests són objectius que volen ser assolits l'any 2020, així doncs són projectes impulsats per la via de finançament europea *Horizon2020 (H2020)*[6]. BIG IoT comença el mes de gener de 2016 i acaba el mes de desembre de 2018.

A BIG IoT es volen assolir els principals objectius de la IoT-EPI mencionats anteriorment; millorar la interoperabilitat entre plataformes IoT, tot impulsant i facilitant el desenvolupament d'aplicacions i serveis que facin ús de dades IoT. També es proposa una estructura central, anomenada *Marketplace*, que emmagatzema informació sobre les dades que es troben actualment al mercat, és a dir, un punt central en què es poden consultar i cercar les dades que ens interessin per tal de desenvolupar la nostra aplicació o servei. Per tal d'aconseguir-ho, BIG IoT proposa diverses eines i estructures:

- Una capa semàntica d'alt nivell, alineada amb els estàndards del *W3C Web of Things group* [13], que permet especificar i descriure els diferents tipus de dades existents al mercat. És la principal eina que permet l'estandardització de les dades.
- Una estructura central que permet i facilita l'intercanvi de dades entre plataformes anomenada *Marketplace*. Utilitza la semàntica del punt anterior per tal de "conèixer" les dades que es troben actualment al mercat, d'aquesta manera el *Marketplace* ens permet cercar i filtrar diferents ofertes de dades segons la seva descripció semàntica.
- Una llibreria en Java, que permet una comunicació homogènia entre el *Marketplace* i les diverses plataformes IoT. Aquesta llibreria permet automatitzar la cerca i obtenció de dades que proporciona el *Marketplace*.

A BIG IoT, i durant aquest treball, s'utilitzarà la següent terminologia per referir-se als diferents components de BIG IoT:

- **offering**: Entitat que representa l'oferiment d'unes dades determinades al Marketplace per part d'un proveïdor de dades, és la unitat bàsica que permet l'automatització dels processos que desenvolupa el *Marketplace*. Aquest *offering* té diversos atributs que defineixen les seves principals característiques; descripció semàntica de les dades, preu d'accés a les dades, llicència, ubicació

de les dades, proveïdor d'aquestes dades i punt d'accés.

- **provider:** Entitat o plataforma que ofereix les seves dades al *Marketplace* mitjançant els *offerings* anteriorment descrits.
- **consumer:** Potencial consumidor de dades.
- **query:** Entitat que representa una petició de cerca de dades al *Marketplace* per part d'un consumidor (*consumer*). Aquesta *query* conté paràmetres que permeten el filtratge dels diferents offerings disponibles al *Marketplace*.
- **servei BIG IoT:** Component software que actua com a pasarel·la entre un proveïdor de dades i el *Marketplace*. Aquest component ofereix les dades al Marketplace, adaptant-les a les descripcions semàntiques que BIG IoT proporciona.

Per exemple, mitjançant BIG IoT, una aplicació consumidora de dades d'aparcament (*consumer*) ja no hauria d'implementar una passarel·la encarregada de comunicar-se amb els diferents proveïdors de dades d'aparcament (*providers*). En canvi, “preguntaria” al *Marketplace* (*query*), mitjançant la llibreria, si existeix alguna oferta de dades d'aparcament (*offering*) que compleixi els requisits demanats (preu, llicència, format o estructura). Un cop s'han identificat les ofertes adients, l'aplicació obtindria les dades mitjançant un servei de subscripcions que facilita BIG IoT. Per tant, utilitzant el *Marketplace* com a intermediari, només és necessari implementar una única passarel·la encarregada de comunicar l'aplicació amb els diferents proveïdors de dades. En aquest cas, el proveïdor de dades les ofereix al *Marketplace* mitjançant un servei BIG IoT.

Així doncs, la solució proposada per BIG IoT es basa en estandarditzar les comunicacions i descripcions de les dades IoT, centralitzant el mercat mitjançant el *Marketplace*. Aquesta solució és compatible amb les infraestructures i plataformes existents, només cal afegir una capa addicional al software existent per tal d'adaptar-lo a BIG IoT. Aquesta capa assegura que l'aplicació o servei serà capaç de treballar amb diferents conjunts de dades, independentment de les característiques del proveïdor.

Al pilot de Barcelona hi col·laboren les empreses Worldsensing[14], Seat[10], l'Ajuntament de Barcelona[1] i la UPC (líder del pilot). Aquestes entitats participen al pilot desenvolupant aplicacions i serveis, o facilitant dades, dins l'ecosistema BIG IoT. Worldsensing proporciona dades d'aparcament a la zona de Les Corts, i dades de trànsit a les rondes. Seat integra, a la seva aplicació de cerca d'aparcament, les dades proporcionades per Worldsensing. L'Ajuntament proporciona dades de la ciutat a través de diverses plataformes. Durant el desenvolupament del projecte s'exposaran amb més detalls les dades que intervenen al pilot.

3 Estat de l'art

Cal insistir en que aquest treball final de grau és part de la prova pilot BIG IoT a Barcelona, l'objectiu de la qual és demostrar i estudiar la viabilitat de les solucions proposades per BIG IoT a la ciutat. La naturalesa d'aquest treball fa que l'estat de l'art sigui particular; en aquest apartat es vol determinar que cap de les alternatives similars a BIG IoT ha sigut implementada a la ciutat de Barcelona i que, per tant, existeix un espai que justifica el desenvolupament d'aquest treball final de grau.

La Comunitat Europea ha impulsat diverses iniciatives que busquen solucionar la problemàtica descrita a la introducció, la *IoT European Platforms Initiative (IoT-EPI)* [8] és un exemple. Aquesta iniciativa agrupa diferents projectes amb objectius similars, BIG IoT i els següents projectes en formen part:

- **symbIoTe** [12] (2016-01-01 / 2019-01-01): En aquest projecte es considera la problemàtica causada per la fragmentació del mercat IoT actual generada per les comunicacions verticals entre plataformes proveïdores i consumidores de dades. Es proposa una capa d'abstracció que permet la interoperabilitat entre plataformes, oferint transparència en les comunicacions. SymbIoTe proposa generar una “vista unificada” de l'escenari actual IoT. Mitjançant la capa d'abstracció es permet la comunicació homogènia entre diverses plataformes IoT, generant així una “federació” de plataformes IoT interoperables entre sí.
- **INTER-IoT** [7] (2016-01-01 / 2018-12-31): Aquest projecte proposa un sistema basat en capes que permet la interoperabilitat entre plataformes de l'entorn IoT. Es proposa una solució alternativa a les integracions a nivell de xarxa dels altres projectes amb objectius similars. Entre les solucions que es desenvolupen hi trobem la interacció a baix nivell entre dispositius, un software que permet la interoperabilitat entre serveis IoT a nivell de xarxa o la definició d'una ontologia per descriure dades heterogènies. També es desenvolupen dues proves pilot relacionades amb casos d'ús d'e-health i logística de transport portuària.
- **bIoTope** [15] (2016-01-01 / 2018-12-31): Aquest projecte proposa una estructura *Systems-of-Systems (SoS)*, que considera cada component de la xarxa com a un servei. La interoperabilitat entre serveis s'aconsegueix mitjançant estàndards de comunicació *Open Messaging Interface (O-MI)* i *Open Data Format (O-DF)*. Per tant, aquest projecte vol incentivar l'ús dels estàndards de comunicació existents, especialment els que permeten interoperabilitat entre Open API's. També es proposa un *framework* que facilita la gestió de la seguretat i privacitat de les dades, a fi de promocionar bones pràctiques en l'ús de dades sensibles. A bIoTope també es porten a terme tres proves pilot a

ciutats europees, Helsinki, Brusselles i Lyon, per tal de demostrar i estudiar la viabilitat de les solucions proposades.

- **Vicinity** [17](2016-01-01 / 2019-12-31): Aquest projecte ofereix un enfoc diferent a la problemàtica derivada de la poca interoperabilitat entre plataformes. Es proposa un “barri virtual”, semblant a una xarxa social, que permet als usuaris configurar els seus dispositius per tal d’interconnectar-los entre sí, fins i tot entre dispositius d’altres usuaris. Aquest mecanisme d’interoperabilitat és conegut com a *Interoperability as a Service*, que permet la interoperabilitat a nivell tècnic de manera transparent a l’usuari. Es proposen dos nivells d’interoperabilitat; nivell de xarxa/infraestructura i nivell de dispositiu. Una de les funcionalitats interessants que proporciona el “barri virtual” és la configuració del nivell de privacitat i seguretat de les dades dels usuaris, proporcionant una solució personalitzada pel que fa al tractament de dades sensibles. En aquest projecte també s’implementa una prova pilot que interconnecta vuit plataformes i dispositius IoT de set països diferents.

Fora de la IoT-EPI també existeixen solucions als reptes que presenta l’escenari IoT actual. Per exemple, existeix Fiware[3]; solució impulsada per empreses com ATOS, Orange i Telefónica. Aquest projecte, ja en producció, ofereix solucions software de codi obert que permeten harmonitzar i estandarditzar les comunicacions entre plataformes IoT. Per tant, l’objectiu principal de Fiware és oferir solucions de codi obert que facilitin el desenvolupament d’aplicacions i serveis que facin ús de dades provinents de diferents fonts. Fiware, com BIG IoT, també ofereix un *Marketplace* que centralitza l’oferta de dades. De totes maneres, Fiware no és del tot lliure, existeix un sistema de quotes anuals que proporciona diferents drets als usuaris de l’ecosistema Fiware en funció de la quantitat pagada.

Els projectes mencionats, i BIG IoT, proposen una solució agnòstica per als dispositius, ja sigui mitjançant estàndards de comunicació o centralitzant les diferents ofertes de dades. No tenim constància que cap dels projectes mencionats anteriorment hagi estat implementat a Barcelona, de manera que es presenta l’oportunitat d’incloure les dades de la ciutat a l’ecosistema de BIG IoT, necessari per a la implementació del pilot BIG IoT a la ciutat.

4 Motivació i objectius

4.1 Motivació

En poques paraules, BIG IoT proposa solucions que faciliten la interoperabilitat entre plataformes proveïdores i consumidores de dades IoT. Per tal de consolidar el projecte, s'implementen tres proves pilot a diverses ciutats d'Europa, amb la finalitat de demostrar la viabilitat de les solucions proposades per BIG IoT.

L'inLab FIB és un laboratori de càlcul que històricament ha donat suport informàtic a la Facultat d'Informàtica de Barcelona (FIB). Cal destacar també que l'inlab FIB integra un departament centrat en la recerca en que s'hi desenvolupen projectes per a clients externs. Les temàtiques de mobilitat i *smart cities* es troben entre les principals àrees d'expertesa de l'inLab, precisament les requerides pel pilot de Barcelona. L'inlab FIB participa activament a BIG IoT liderant el pilot de Barcelona juntament amb el paquet de treball que coordina la resta de pilots del projecte. És necessari coordinar els interessos de les diferents parts implicades a fi de que cada una d'aquestes aporti el màxim valor possible al pilot a desenvolupar.

Per tal de realitzar la prova pilot a Barcelona es requereix una mostra representativa dels diferents actors de la ciutat, especialment els proveïdors de dades. L'Ajuntament de Barcelona (com a representat de l'administració), Worldsensing (com a representant d'empresa emergent) i Seat (com a representant de empresa consolidada). La coordinació i integració dels interessos particulars de cada col·laborador és també una part essencial de la prova pilot. Actualment aquestes empreses proporcionen dades de la ciutat mitjançant un sistema de comunicació amb els consumidors completament vertical i independent. Per tant, l'adopció de les solucions proposades per BIG IoT presenta una nova prova de concepte per aquestes empreses, interessades en obtenir els beneficis que proporciona BIG IoT.

Com que BIG IoT és un projecte impulsat per la Comunitat Europea, un dels principals objectius és la millora de la qualitat de vida dels ciutadans des d'una perspectiva tecnològica i innovadora. BIG IoT facilita i incentiva l'accés a dades, que poden ser utilitzades per millorar diversos aspectes de la vida quotidiana dels ciutadans. Atès que les dades disponibles per desenvolupar el pilot a Barcelona estan relacionades amb l'àmbit de la mobilitat i les *smart cities*, es presenta l'oportunitat d'idear casos d'ús relacionats amb aquesta temàtica.

Per al pilot, es contempla l'escenari en que existeix una persona encarregada de la gestió de la mobilitat a la ciutat. Aquesta persona té accés a informació proporcionada per diverses fonts, tant públiques com privades, que és utilitzada per millorar les decisions dels ciutadans a l'escollir la manera de moure's per la ciutat.

És a dir, en aquest escenari les aplicacions i serveis utilitzats pels usuaris integren els beneficis que proporciona BIG IoT: interoperabilitat entre diferents plataformes, homogeneïtat en les comunicacions i facilitat en la cerca de nous proveïdors de dades, entre d'altres.

Així doncs, es presenta l'oportunitat de realitzar un treball final de grau en que es desenvoluparan part de les aplicacions i serveis requerits pel pilot de Barcelona relacionats amb les temàtiques de mobilitat i *smart cities*. A continuació es detallen els objectius del treball.

4.2 Objectius

L'objectiu principal és desenvolupar serveis i aplicacions que proveeixin i consumeixin dades de la ciutat de Barcelona utilitzant les eines i l'ecosistema proporcionats per BIG IoT. Aquests serveis i aplicacions interactuaran amb els desenvolupats pels altres membres del pilot. La demostració i la justificació del pilot seran realitzades mitjançant casos d'ús relacionats amb l'àmbit de la mobilitat.

Més concretament, els principals objectius i components a desenvolupar del treball són:

- Desenvolupar serveis BIG IoT que ofereixin al *Marketplace* les dades obertes de Barcelona, proporcionades per la plataforma *Open Data Bcn* [9].
- Mantenir i actualitzar els serveis BIG IoT desenvolupats per *Worldsensing* encarregats de proveir dades d'aparcament i flux de trànsit.
- Desenvolupar el *Traffic Information Center Tool*, panell de control en què es mostraran dades de la ciutat de Barcelona. Aquest panell té com a objectiu visualitzar diferents tipus de dades per tal d'ajudar en la presa de decisions relacionades amb la mobilitat.
- Desenvolupar la *BCN Traffic Info App*, aplicació mòbil per al ciutadà relacionada amb l'àmbit de la mobilitat. Aquesta aplicació ha de d'incentivar al ciutadà moure's de manera sostenible per la ciutat.
- Desenvolupar un sistema d'alarmes i recomanacions que avisin i ajudin al ciutadà a prendre decisions relacionades amb la mobilitat.

5 Abast

L'abast d'aquest projecte se centra en el desenvolupament dels diferents components requerits pel pilot BIG IoT a la ciutat de Barcelona; els serveis BIG IoT, l'aplicació web *Traffic Information Center Tool* i l'aplicació mòbil *BCN Traffic Info*.

És necessari recalcar que les aplicacions i serveis desenvolupats al pilot de Barcelona, relacionats amb l'àmbit de la mobilitat i les *smart cities*, no busquen competir amb les ja establertes al mercat actual. Cal recordar que l'objectiu principal és demostrar i estudiar la viabilitat de les solucions proposades pel projecte BIG IoT a la ciutat, mitjançant casos d'ús relacionats amb la mobilitat i les *smart cities*.

La Figura 1 mostra la vista general dels components a desenvolupar. A continuació es descriuen els diferents serveis i aplicacions.

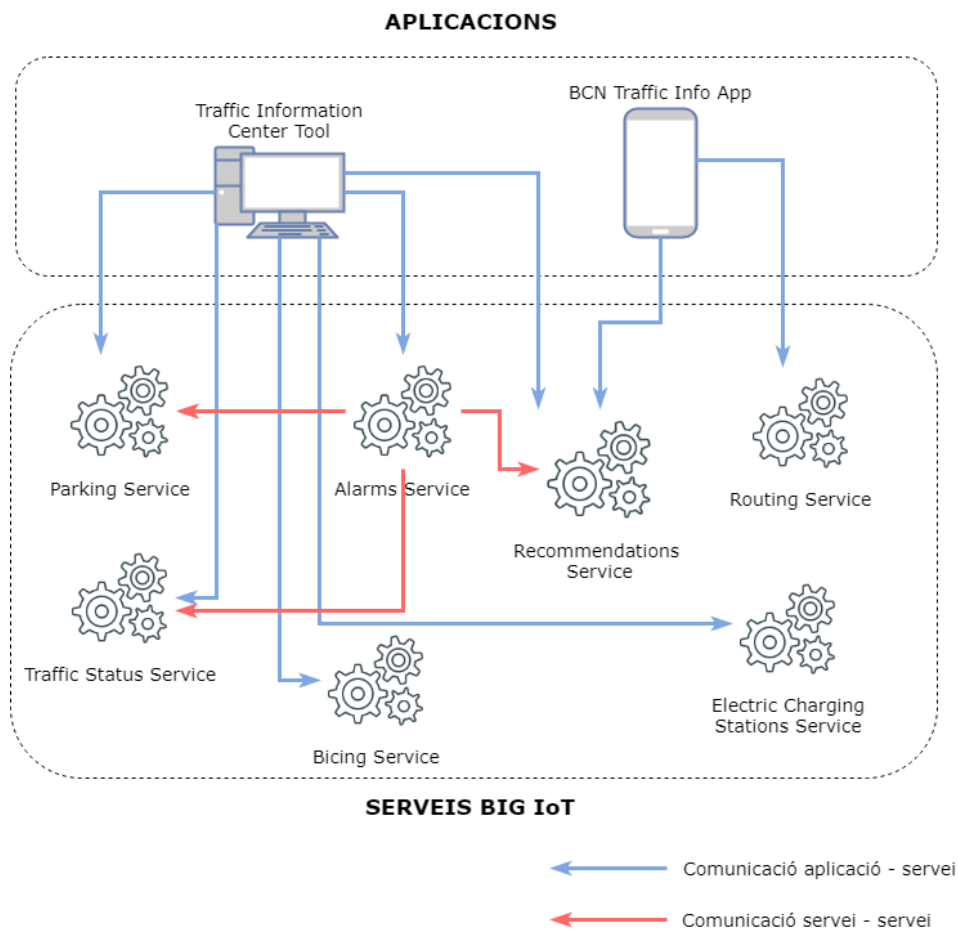


Figura 1: Vista general dels components a desenvolupar

Serveis BIG IoT

Els serveis BIG IoT són els components encarregats de la integració de les dades d'un proveïdor amb el *Marketplace* (i l'ecosistema BIG IoT). Per a la prova pilot de Barcelona, i aquest treball, es requereixen dades provinents de diverses fonts, de manera que per a cada una d'elles es desenvolupa un servei BIG IoT.

Per tal d'oferir les dades al *Marketplace*, el servei BIG IoT necessita obtenir les dades de la font i adaptar-les a la capa semàntica adient proporcionada per BIG IoT.

En aquest treball es desenvoluparan els següents serveis, cada un d'ells encarregat d'oferir un tipus de dades al Marketplace. A la secció d'especificació 8.4 es detallen les funcionalitats de cada servei BIG IoT.

- **Parking Service:** Servei BIG IoT encarregat d'oferir al *Marketplace* les dades d'aparcament de Les Corts proporcionades per l'empresa Worldsensing.
- **Traffic Status Service:** Servei BIG IoT encarregat d'oferir al *Marketplace* les dades de trànsit de les Rondes, també proporcionades per Worldsensing.
- **Electric Charging Stations Service:** Servei BIG IoT encarregat d'oferir les dades d'estacions elèctriques proporcionades per la plataforma OpenData BCN de l'Ajuntament de Barcelona.
- **Bicing Service:** Servei BIG IoT encarregat d'oferir les dades del servei de bicis compartides, *Bicing*, de l'Ajuntament de Barcelona. Les dades també són extretes de la plataforma OpenData BCN.
- **Routing Service:** Servei BIG IoT encarregat de retornar la ruta i temps de viatge entre dos punts de la ciutat en diferents modes de transport (cotxe, en bici o caminant). Servei necessari per tal d'implementar l'aplicació mòbil *BCN Traffic Info App*, que vol incentivar la mobilitat sostenible del ciutadà.
- **Recommendations Service:** Servei BIG IoT encarregat de gestionar les recomanacions de la ciutat. Aquestes recomanacions són la eina que permet al gestor de la mobilitat enviar informació als ciutadans, que rebran les recomanacions a l'aplicació mòbil.
- **Alarms Service:** Servei BIG IoT encarregat de gestionar les alarmes, que permeten al gestor de la mobilitat monitorar les dades d'aparcament a Les Corts i velocitat de trànsit a les Rondes. Aquest servei també permet activar i desactivar recomanacions automàticament en funció de les dades que estan sent monitorades. A la secció 9.1.1 s'hi detallen l'arquitectura i funcionalitats, ja que per tal de monitorar les dades, aquest servei integra un sistema de monitoratge extern.

Tal i com es pot veure a la Figura 1 els serveis proporcionen dades a les aplicacions web i mòbil. A la figura també s’hi poden apreciar les comunicacions entre serveis, aquestes comunicacions també es realitzen mitjançant les eines que proporciona BIG IoT.

Concretament, el servei d’alarmes obté les dades a monitorar d’aparcament i velocitat mitjançant la llibreria BIG IoT, tal i com ho realitzaria un consumidor de dades normal.

Traffic Information Center Tool

Aplicació web pensada perquè sigui utilitzada per un gestor de la mobilitat de la ciutat. Aquesta aplicació mostra les diferents dades del pilot de Barcelona: aparcament, trànsit, estacions de càrrega de vehicles elèctrics i servei *Bicing*. Cada un d’aquest grup de dades és extret del corresponent servei BIG IoT.

Aquesta aplicació també permet al gestor enviar recomanacions als ciutadans, relacionades amb la mobilitat. Aquestes recomanacions són la principal via de comunicació entre el gestor de la mobilitat i els ciutadans, ja que aquests les reben a l’aplicació mòbil *BCN Traffic Info*. Les recomanacions estan formades per un nom, descripció i àrea d’afectació.

L’aplicació també permet a l’usuari gestionar les alarmes, pensades per monitorar dades d’aparcament a Les Corts i velocitat a les Rondes.

En cas de que les dades a monitorar siguin les de velocitat, l’aplicació permet escollir un límit de velocitat i un operand (major, menor o igual). Els trams de les Rondes que compleixin la condició establerta seran monitorats pel sistema d’alarmes. En cas de que les dades siguin d’aparcament, l’aplicació permet a l’usuari escollir espais d’aparcament lliures o ocupats. En els apartats d’Especificació i Disseny es detallen més concretament les funcionalitats que ofereix el servei d’alarmes.

Tal i com es pot veure, aquesta aplicació web utilitza diferents serveis BIG IoT per tal de realitzar les funcionalitats requerides.

BCN Traffic Info App

Aplicació mòbil pensada per ser utilitzada pels ciutadans de la ciutat de Barcelona, que tenen com a objectiu millorar la sostenibilitat de la seva mobilitat per la ciutat.

Aquesta aplicació permet als usuaris consultar la ruta i temps de viatge entre dos punts, tenint en compte i informant de les recomanacions actives trobades a la ruta

demanada.

També es té en compte el mode de transport de la ruta i, en cas que sigui necessari, es proposen rutes alternatives amb modes de transport més sostenibles, ja siguin en bici o caminant.

5.1 Possibles obstacles

Dependència dels artefactes BIG IoT

El mal funcionament dels artefactes desenvolupats per BIG IoT (*Marketplace*, llibreria o ontologia) podria ser un greu obstacle, ja que impossibilitaria la correcta implementació dels pilots i, per tant, d'aquest treball final de grau.

Dependència de fonts de dades externes

Les dades que s'utilitzen en aquest treball final de grau provenen de fonts externes que escapen del nostre control. En cas de que aquestes fonts presentin problemes, el desenvolupament dels components que facin ús d'aquestes dades es pot veure afectat.

Alteracions en la planificació temporal

El pilot ha de ser promocionat als futurs esdeveniments que es presentin durant el desenvolupament del projecte. Per tant, la planificació es pot veure alterada per tal de preparar les demostracions requerides.

Adaptació de la metodologia de treball

Tal i com s'exposa a l'apartat de metodologies, es vol treballar amb metodologies àgils. La manca del rol de client convencional, *product owner*, farà que s'hagi d'adaptar la metodologia d'acord amb la naturalesa i context del projecte.

6 *Stakeholders*

A continuació es detallen els diferents actors implicats, *stakeholders*, d'aquest projecte. Cal diferenciar dos tipus d'*stakeholders*: els del pilot de Barcelona, amb interessos particulars, i els usuaris finals de les aplicacions i serveis a desenvolupar.

- **Partners del pilot de Barcelona**

Worldsensing: Startup especialitzada en sensors i processat de dades IoT. Col·labora al pilot proporcionant dades de trànsit de les Rondes i d'aparcament a la zona de Les Corts. Té interès en oferir part de les seves dades al pilot de Barcelona per tal de realitzar una prova de concepte amb les eines que proporciona BIG IoT. Aquesta prova de concepte és beneficiosa per l'empresa ja que està directament relacionada amb el seu model de negoci.

Ajuntament de Barcelona: Administració que proporciona dades obertes relacionades amb la mobilitat de la ciutat. Pel projecte s'utilitzen les dades del servei *Bicing* i informació de les diferents estacions de càrrega elèctriques distribuïdes per la ciutat. L'administració està interessada en oferir dades obertes al pilot per tal d'estudiar els beneficis que ofereix BIG IoT.

Seat: Empresa automobilística. Col·labora al pilot proporcionant dades de contaminació, recol·lectades per un reduït grup de cotxes que circulen per la ciutat. Aquesta empresa també està interessada en estudiar els beneficis que ofereix BIG IoT a fi de millorar la gestió de les dades que ofereixen els seus automòbils.

- **Usuaris**

Usuari administració: Treballador de l'administració que desenvolupa funcions relacionades en la presa de decisions en l'àmbit de la mobilitat de la ciutat. Utilitza informació i eines proporcionades pel panell de control *Traffic Information Center Tool* per tal de millorar la mobilitat dels ciutadans per la ciutat.

Usuari ciutadà:Usuari amb accés a l'aplicació mòbil *BNC Traffic Info*. L'usuari utilitza aquesta aplicació amb l'objectiu de millorar la seva mobilitat per la ciutat.

També cal destacar que es considera **BIG IoT** (o el conjunt dels seus col·laboradors) com a principal *stakeholder* d'aquest treball, interessat en que el pilot de Barcelona assoleixi els objectius amb èxit, demostrant així que BIG IoT seria viable per la ciutat de Barcelona. En última instància, també cal tenir en compte els **promotors** d'aquest treball: l'inlab FIB i l'estudiant interessat en dur-lo a terme, jo mateix.

7 Metodologia i rigor

Per al desenvolupament d'aquest treball final de grau s'utilitzaran metodologies àgils. Cal recordar que l'equip participa al projecte europeu BIG IoT de manera activa, en el que la principal metodologia utilitzada és la clàssica; tres grans iteracions durant la fase de desenvolupament del projecte. Al final de cada iteració s'entrega als revisors europeus la documentació en què s'exposa i presenta la feina realitzada durant la iteració.

L'equip de treball de l'inLab FIB és format per les següents persones: Dr. Josep Casanovas com a investigador principal del projecte, Dra. Mari Paz Linares com a cap del pilot de Barcelona i Dr. Ernest Teniente com a cap dels pilots de BIG IoT. L'equip de desenvolupament és format per en Juan Salmerón, el Dr. Xavier Oriol i l'Àlex Català.

Per al desenvolupament intern de les tasques relacionades amb el pilot trobem que les metodologies àgils s'adeqüen més a l'equip i a la manera de treballar. De totes maneres, la metodologia que s'utilitzarà finalment serà subjecta a les màximes que es derivin de la metodologia clàssica emprada a BIG IoT.

7.1 Metodologies de desenvolupament

D'acord amb el que s'ha mencionat anteriorment, es treballarà mitjançant metodologies àgils, que plantegen una alternativa a l'altament emprada metodologia clàssica. La idea predominant d'aquest tipus de metodologies és el desenvolupament iteratiu i incremental, en que els requisits i solucions es poden veure modificats durant el desenvolupament del projecte. Existeixen moltes variants de les metodologies àgils, ja que poden ser adaptades a diferents equips o empreses. En aquest treball s'ha utilitzat la següent metodologia:

Les tasques a desenvolupar s'organitzen en iteracions (*sprints*) de dues setmanes. L'equip es reuneix al final d'*sprint* per tal de revisar la feina completada durant la iteració. També es realitzen reunions diàries d'equip (*dailys*), útils per tal d'agilitzar el desenvolupament i desencallar tasques que requereixen coneixements d'altres membres de l'equip. Cada dos mesos també es realitza una reunió amb tots els membres de l'equip anomenada *retrospective meeting*, centrada en analitzar i enfortir les relacions de treball entre els integrants de l'equip.

Per tal de facilitar el desplegament dels serveis i aplicacions s'utilitzarà *Integració Contínua*. Aquesta metodologia ens proporciona un entorn que, a cada nova versió

del software desenvolupat, genera un artefacte automàticament capaç de ser desplegat a producció.

7.2 Control de gestió

Al final de cada tasca es valorarà si els recursos destinats corresponen amb els estimats. En cas de que existeixin desviacions majors, s'analitzaran les causes o raons per tal d'evitar cometre el mateix error en futures tasques. Alguns exemples de desviacions que es poden presentar:

- Error en l'estimació temporal d'una tasca.
- Error en la definició dels requisits i objectius d'una tasca.
- No poder finalitzar una tasca per motius aliens a l'equip.

El fet de treballar amb metodologies àgils ens permet modificar i corregir errors de futures iteracions, minimitzant així les desviacions menors en futures tasques.

Si el pressupost final és superior a l'estimat, s'haurà d'assignar la part reservada a l'apartat de contingències per tal d'assumir els costos extrems.

7.3 Valoració d'alternatives i pla d'acció

Les funcionalitats i requisits d'aquest projecte es deriven de BIG IoT i els diferents col·laboradors del projecte. De totes maneres, aquests requisits són suficientment laxos com per permetre un marge a l'hora de prendre decisions; ja siguin a nivell tècnic o per incloure funcionalitats.

Aquest marge permet que en la planificació es puguin afegir noves tasques, derivades de possibles nous requisits, i treure'n d'altres que afegeixen valor al pilot però que no varen sorgir explícitament d'algun requisit.

És a dir, l'objectiu principal del projecte és demostrar que BIG IoT és viable a la ciutat de Barcelona, per tant, aquest objectiu proporciona un marge de maniobra que permet modificar la planificació, a nivell de tasques, sense que l'objectiu final quedi afectat.

Si apareixen obstacles que involucren a terceres parts, ja siguin col·laboradors del pilot o projecte europeu, s'estudiaran possibles noves solucions d'acord amb els nous requisits i/o restriccions plantejats.

7.4 Eines de seguiment

Pel correcte seguiment del projecte s'han utilitzat les següents eines:

- Trello: Taulell virtual que ens permet especificar, descriure i organitzar les diferents tasques i histories d'usuari.
- Git[4]: Tot el codi del projecte s'emmagatzema a la plataforma GitLab[5] gràcies al control de versions que ens proporciona Git.
- Slack: Versàtil eina de comunicació utilitzada entre els membres de l'equip de desenvolupament.

8 Especificació

En aquest apartat s'identifiquen i especifiquen els objectius dels *stakeholders* i requisits que se'n deriven. També es detallen les funcionalitats i models conceptuals de les aplicacions i serveis plantejats.

8.1 Objectius de les parts interessades

Els *stakeholders* mencionats anteriorment tenen interessos en el correcte desenvolupament del pilot de Barcelona. D'aquests interessos se'n deriven objectius concrets que posteriorment donen lloc als requisits funcionals i no funcionals.

Objectiu 1

Promotors. El projecte a desenvolupar serà vàlid com a treball final de grau.

Objectiu 2

Promotors. El cost del treball s'adequarà a la part corresponent del finançament que rep l'inlab FIB per tal de desenvolupar el pilot BIG IoT a Barcelona.

Objectiu 3

BIG IoT. El software desenvolupat haurà d'utilitzar les eines i ecosistemes proporcionats pel projecte europeu BIG IoT.

Objectiu 4

BIG IoT. Les aplicacions i serveis desenvolupats estaran disponibles durant el desenvolupament del treball per tal de realitzar demostracions.

Objectiu 5

Worldsensing. Les dades de trànsit i aparcament seran integrades a l'ecosistema BIG IoT.

Objectiu 6

Worldsensing. Les dades de trànsit i aparcament tindran visibilitat a les aplicacions a desenvolupar.

Objectiu 7

Ajuntament de Barcelona. Les dades del servei *Bicing* i les d'estacions de càrrega elèctrica seran integrades a l'ecosistema BIG IoT.

Objectiu 8

Ajuntament de Barcelona. Les dades del servei *Bicing* i les d'estacions de càrrega elèctrica tindran visibilitat a les aplicacions a desenvolupar.

Objectiu 9

Seat. Les dades de contaminació proporcionades tindran visibilitat a les aplicacions a desenvolupar.

Objectiu 10

Usuari administració. Mitjançant el panell de control l'usuari podrà visualitzar les dades relacionades amb la mobilitat de la ciutat de Barcelona. També podrà crear alarmes que monitoraran la disponibilitat dels aparcaments a la zona de Les Corts o la velocitat de les Rondes. Aquestes alarmes podran crear recomanacions automàticament en cas de que les dades monitorades presentin alguna anomalia. L'usuari també podrà crear manualment recomanacions.

Objectiu 11

Usuari administració. L'usuari podrà crear alarmes que monitoraran la disponibilitat dels aparcaments a la zona de Les Corts o la velocitat de les Rondes. Aquestes alarmes podran crear recomanacions automàticament en cas de que les dades monitorades presentin alguna anomalia. L'usuari també podrà crear manualment recomanacions.

Objectiu 12

Usuari ciutadà. L'usuari podrà obtenir rutes i temps de viatge entre dos punts de la ciutat de Barcelona en tres modes de transport diferents: cotxe, bici o caminant. En cas de que el mode triat sigui el cotxe, l'aplicació proporcionarà rutes alternatives per tal d'incentivar una mobilitat sostenible.

Objectiu 13

Usuari ciutadà. L'usuari podrà visualitzar les recomanacions actives a la ciutat de Barcelona. També podrà visualitzar les recomanacions actives incloses en la ruta demanada.

8.2 Requisits no funcionals

Usabilitat

- **Descripció:** L'aplicació ha de ser intuïtiva, còmoda i usable pels usuaris.
- **Justificació:** Les aplicacions poden ser utilitzades per persones poc expertes en l'àmbit TIC.

Manteniment i disponibilitat

- **Descripció:** Els components desenvolupats empraran tecnologies de programari lliure i no utilitzaran llibreries de tercers que tinguin associades llicències que en limitin l'ús.
- **Justificació:** El codi desenvolupat mitjançant tecnologies de programari lliure és més fàcil de mantenir. El fet de que els components formin part d'un projecte europeu també implica que el codi a desenvolupar sigui obert.

Ús d'eines i ecosistema BIG IoT

- **Descripció:** Els components desenvolupats hauran d'utilitzar les eines i ecosistema proporcionats per BIG IoT.
- **Justificació:** La interacció entre els diferents components cal que es realitzi mitjançant BIG IoT per tal de demostrar la viabilitat de les solucions proposades pel projecte europeu.

Arquitectura orientada a serveis

- **Descripció:** Organitzar les diferents funcionalitats en serveis que permeten una solució modularitzada.
- **Justificació:** Les eines i ecosistema BIG IoT utilitzats incentiven una arquitectura modular en que es minimitzen les dependències de cada component i se'n facilita la interoperabilitat. Per tant, és necessari estructurar una arquitectura d'aquestes característiques per tal de dur a terme els objectius del pilot de Barcelona i, en conseqüència, d'aquest treball.

8.3 Requisits funcionals

A continuació es detallen els requisits funcionals d'aquest treball, distribuïts en dos grups: els relacionats amb el *Traffic Information Center Tool*, els de la *BCN Traffic Info*.

8.3.1 *Traffic Information Center Tool*

Visualitzar dades

- **Descripció:** L'aplicació ha de mostrar les dades ofertes pels col·laboradors del pilot. Cal que les dades es mostrin en temps real.
- **Justificació:** És necessari mostrar les dades relacionades amb la mobilitat de Barcelona per tal de demostrar la correcta integració de l'ecosistema BIG IoT als components desenvolupats.

Gestió alarmes i recomanacions

- **Descripció:** L'aplicació ha de proporcionar un sistema d'alarmes i recomanacions basat en el monitoratge de dades d'aparcament i trànsit.
- **Justificació:** És necessari incloure al pilot un component que interaccioni de manera complexa amb els proveïdors per tal de demostrar que l'ecosistema BIG IoT ho permet.

8.3.2 *BCN Traffic Info*

Consultar rutes i temps de viatge

- **Descripció:** L'aplicació ha de permetre a l'usuari consultar la ruta i temps de viatge entre dos punts de la ciutat, oferint tres modes de transport diferents: cotxe, bici o caminant. També ha de mostrar informació relativa a les rutes que afectin la ruta demanada. L'aplicació també ha de proporcionar rutes verdes, en bici o caminant, en cas que el mode de transport principal sigui el cotxe.
- **Justificació:** Per tal de millorar la mobilitat del ciutadà és necessari que les recomanacions estiguin integrades en l'algoritme que permet obtenir la ruta demanada. Aquesta mobilitat també millorarà si és sostenible i el ciutadà fa

menys ús de modes de transport com el cotxe.

Consultar recomanacions

- **Descripció:** El sistema ha de proporcionar la visualització de totes les recomanacions actives a la ciutat de Barcelona.
- **Justificació:** El fet d'obtenir informació de totes les recomanacions actives a la ciutat facilita i millora la presa de decisions del ciutadà.

8.4 Funcionalitats

En aquest punt cal tenir en compte el requisit no funcional que obliga a implementar una arquitectura orientada a serveis (quart requisit de l'apartat 8.2). És per això que es plantegen set serveis encarregats de realitzar les principals regles de negoci requerides. Aquests serveis utilitzaran l'ecosistema BIG IoT per tal de realitzar les diverses funcionalitats, durant aquest treball seran anomenats serveis BIG IoT.

A continuació es detallen les funcionalitats dels serveis BIG IoT, les del panell de control *Traffic Information Center Tool* i les de l'aplicació mòbil *BCN Traffic Info*.

8.4.1 Serveis BIG IoT

Seguidament es detallen les principals funcionalitats dels serveis BIG IoT a desenvolupar.

Parking Service

Aquest servei ha d'oferir al *Marketplace* les dades d'aparcament que proporciona l'*API* de Worldsensing. Les dades ofertes han d'estar sincronitzades en temps real amb les dades de l'*API* de Worldsensing. El format i estructura de les dades ofertes al *Marketplace* han de concordar amb les descripcions semàntiques proporcionades per BIG IoT.

Traffic Status Service

Aquest servei ha d'oferir al *Marketplace* les dades de trànsit que proporciona l'*API* de Worldsensing. Les dades ofertes han d'estar sincronitzades en temps real amb les dades de l'*API* de Worldsensing. El format i estructura de les dades ofertes al *Marketplace* han de concordar amb les descripcions semàntiques proporcionades per

BIG IoT.

Electric Charging Stations Service

Aquest servei ha d'oferir al *Marketplace* les dades d'estacions de càrrega elèctriques proporcionades per la plataforma Open Data BCN. Les dades a retornar han d'estar sincronitzades en temps real amb les dades retornades per la plataforma Open Data BCN. El format i estructura de les dades ofertes al *Marketplace* han de concordar amb les descripcions semàntiques proporcionades per BIG IoT.

Bicing Service

Aquest servei ha d'oferir al *Marketplace* les dades del servei de Bicing proporcionades per la plataforma Open Data BCN. Les dades a retornar han d'estar sincronitzades en temps real amb les dades retornades per la plataforma Open Data BCN. El format i estructura de les dades ofertes al *Marketplace* han de concordar amb les descripcions semàntiques proporcionades per BIG IoT.

Routing Service

Aquest servei ha d'oferir al *Marketplace* la funcionalitat de retornar la ruta entre dos punts tenint en compte el mode de transport sol·licitat (cotxe, bici o caminant). Per requisits del projecte europeu BIG IoT, el proveïdor d'aquesta informació ha de ser extern al pilot de Barcelona. Serà proporcionada pel sistema Open Trip Planner facilitat pels col·laboradors de BIG IoT de Piedmont, Itàlia. El servei també farà ús de les descripcions semàntiques proporcionades per BIG IoT.

Recommendations Service

Aquest servei ha d'oferir al *Marketplace* les funcionalitats requerides per a la gestió de recomanacions: creació, esborrat, activació i desactivació. Aquestes recomanacions han de tenir associada un àrea d'afectació, un nom, una descripció i, en cas que sigui necessari, un llistat de modes de transport restringits a l'àrea d'afectació. Aquestes recomanacions poden estar actives o inactives. El servei també farà ús de les descripcions semàntiques proporcionades per BIG IoT.

Alarms Service

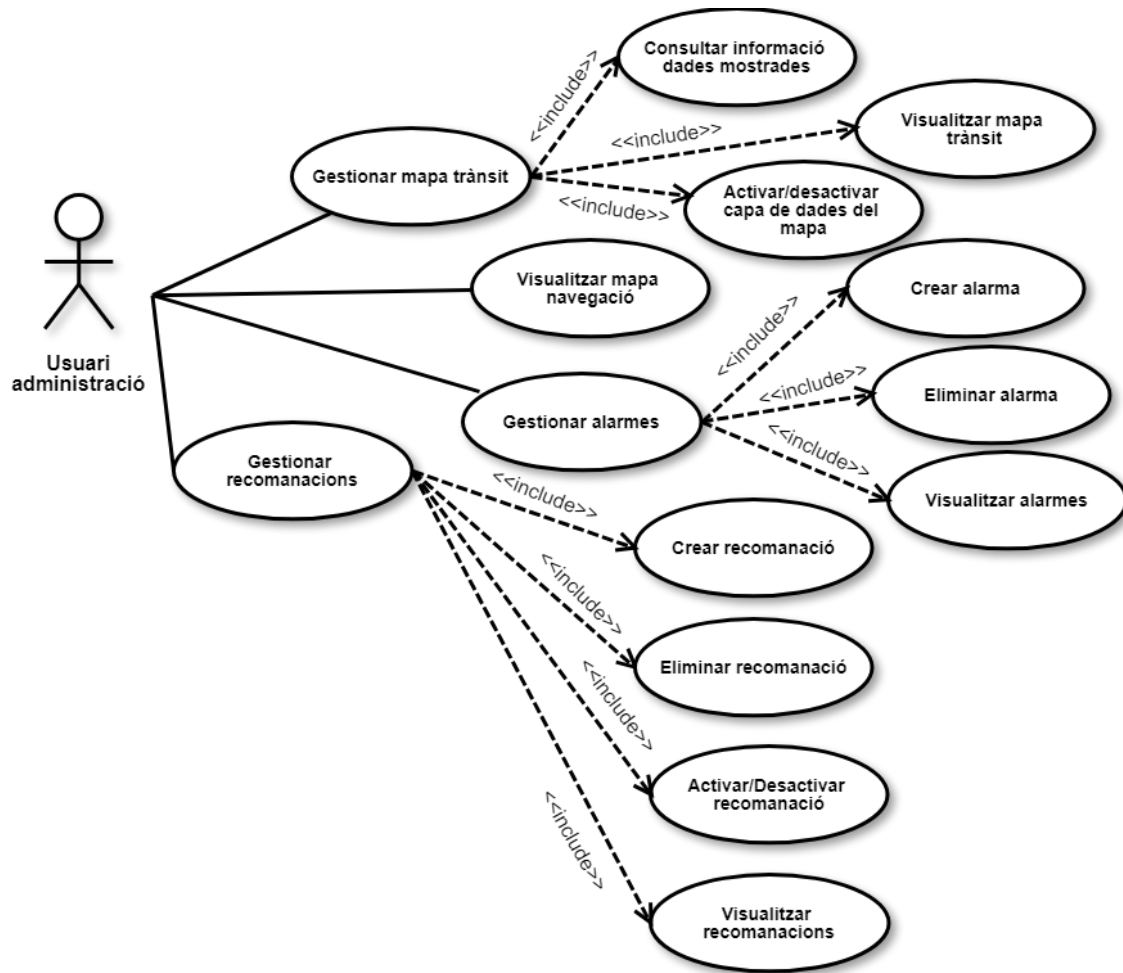
Aquest servei ha d'oferir al *Marketplace* les funcionalitats requerides per a la gestió d'alarmes: creació i esborrat. També ha d'implementar un sistema de monitoratge de la disponibilitat dels aparcaments a la zona de Les Corts i de velocitat del trànsit a les Rondes. També ha de permetre associar recomanacions que s'activin i desactivin automàticament en funció de l'estat dels valors que s'estan monitorant.

Per tant, les alarmes tindran un nom, una descripció, uns valors a monitorar i unes recomanacions associades. El servei també farà ús de les descripcions semàntiques proporcionades per BIG IoT.

8.4.2 *Traffic Information Center Tool*

A continuació es descriuen les funcionalitats del panell de control per a la mobilitat de Barcelona, el *Traffic Information Center Tool*. L'objectiu principal d'aquest panell de control és mostrar diverses dades de la ciutat. També és important que es mostri informació extra relacionada amb els proveïdors de dades, obtinguda pel fet d'utilitzar l'ecosistema BIG IoT. També s'inclouen funcionalitats relacionades amb la gestió d'alarmes i recomanacions facilitada pels corresponents serveis BIG IoT.

Diagrama casos d'ús

Figura 2: Diagrama de casos d'ús *Traffic Information Center Tool***Visualitzar mapa trànsit**

El sistema ha de permetre a l'usuari visualitzar un mapa en que es mostrin les dades de mobilitat de la ciutat en temps real. Aquest mapa ha d'integrar les dades d'aparcament a Les Corts, de l'estat del trànsit a les Rondes, del servei de Bicing i de les estacions de càrrega elèctrica de Barcelona que proporcionen els corresponents serveis BIG IoT.

Consultar informació dades mostrades

El sistema ha de permetre a l'usuari visualitzar la informació bàsica sobre els proveïdors de les dades mostrades. Aquesta informació és fàcilment accessible gràcies a

l'ecosistema BIG IoT.

Activar/desactivar capa de dades del mapa

Les diverses dades han de ser mostrades mitjançant marcadors diferenciats per tal de que l'usuari pugui triar el conjunt de dades que vol visualitzar.

Visualitzar alarmes

El sistema ha de permetre a l'usuari visualitzar les alarmes que es troben actives al sistema. Aquesta visualització ha de mostrar les principals característiques de l'alarma: nom, descripció i un mapa amb els valors resultants del monitoratge. També ha de mostrar el nom de les recomanacions associades.

Crear alarma

El sistema ha de permetre a l'usuari crear una alarma. El sistema ha de permetre a l'usuari introduir, mitjançant un formulari, les principals característiques d'una alarma. El sistema ha d'evitar que es l'usuari pugui crear alarmes inconsistents.

Eliminar alarma

El sistema ha de permetre a l'usuari eliminar una alarma existent al sistema.

Visualitzar recomanacions

El sistema ha de permetre a l'usuari visualitzar les recomanacions existents al sistema. Per cada recomanació s'han de mostrar les característiques bàsiques: nom, descripció i àrea d'afectació. Aquesta última ha de ser mostrada en un petit mapa per a cada recomanació.

Crear recomanació

El sistema ha de permetre a l'usuari crear una recomanació. El sistema ha de permetre a l'usuari introduir, mitjançant un formulari, les principals característiques d'una recomanació. El sistema ha d'evitar que es l'usuari pugui crear recomanacions inconsistents.

Eliminar recomanació

El sistema ha de permetre a l'usuari eliminar una recomanació existent al sistema.

Activar/desactivar recomanació

El sistema ha de permetre a l'usuari activar/desactivar una recomanació existent al sistema. El sistema ha de diferenciar clarament recomanacions activades i desactivades per tal d'evitar accions errònies per part de l'usuari.

Visualitzar mapa navegació

El sistema ha de permetre a l'usuari visualitzar les dades de contaminació de la ciutat, proporcionades pels vehicles equipats amb sensors de contaminació de Seat. El mapa ha de mostrar la localització actual de l'usuari.

8.4.3 *BCN Traffic Info*

A continuació es descriuen les funcionalitats de l'aplicació mòbil *BCN Traffic Info*. Aquesta aplicació permet a l'usuari obtenir la ruta i temps de viatge entre dos punts de la ciutat de Barcelona. En cas de que el mode de transport sigui el cotxe es proporcionen rutes alternatives en bici o caminant per tal d'incentivar una mobilitat sostenible. L'aplicació també mostra les recomanacions que afectin la ruta demanada. També es poden visualitzar totes les recomanacions actives a la ciutat.

Diagrama casos d'ús

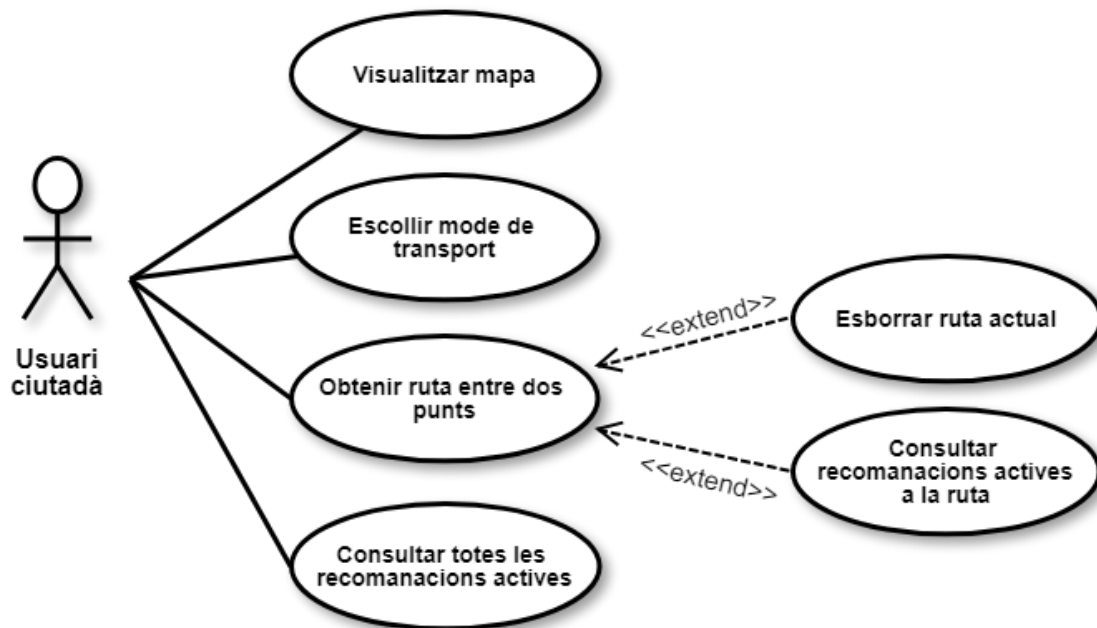


Figura 3: Diagrama de casos d'us *BCN Traffic Info*.

Visualitzar mapa

El sistema ha de permetre a l'usuari visualitzar el mapa de la ciutat de Barcelona, vista en que podrà realitzar les funcionalitats relacionades amb l'obtenció de rutes i temps de viatge.

Obtenir ruta entre dos punts

El sistema ha de permetre a l'usuari introduir dos punts de la ciutat de Barcelona i mostrar la ruta i temps de viatge obtinguts. El sistema també ha de mostrar rutes alternatives en bici o caminant en cas de que el mode de transport seleccionat sigui el cotxe. En cas de que existeixi una ruta alternativa, el sistema ha d'informar clarament a l'usuari de que existeix una ruta alternativa més sostenible.

Escollir mode de transport

El sistema ha de permetre a l'usuari escollir entre tres modes de transport a l'hora d'obtenir la ruta: caminant, en bici o en cotxe.

Esborrar ruta actual

El sistema ha de permetre a l'usuari esborrar la ruta actual per tal de poder introduir noves peticions.

Consultar recomanacions actives a la ruta

En cas de que existeixin recomanacions actives que afectin la ruta demanada, el sistema ha d'informar clarament a l'usuari de que existeixen aquestes recomanacions. El sistema també ha de permetre, a l'usuari, visualitzar aquestes recomanacions.

Consultar totes les recomanacions actives

El sistema ha de permetre a l'usuari visualitzar totes les recomanacions actives a la ciutat de Barcelona.

8.5 Model conceptual

El següent model conceptual il·lustra els conceptes necessaris per dur a terme les funcionalitats descrites.

Dades BIG IoT és la classe principal d'aquest diagrama, qualssevol dada intercanviada entre serveis BIG IoT és considerada *Dada BIG IoT*. Per tant, tant les diverses dades de mobilitat com les rutes, recomanacions i alarmes són Dades BIG IoT.

En aquest punt cal recordar el funcionament de BIG IoT; les ofertes de dades per part dels proveïdors són la entitat clau que permet la comunicació amb els consumidors. Aquest concepte és modelitzat mitjançant la classe *Oferta dades*. Aquesta oferta de dades és visible al *Marketplace* i està descrita per uns paràmetres d'entrada i sortida que segueixen la semàntica BIG IoT.

Els paràmetres d'entrada són aquells que necessita l'oferiment de dades per tal de retornar les dades demanades. Per exemple, el servei que proporciona la ruta entre dos punts necessita que la oferta al *Marketplace* tingui paràmetres d'entrada que representin els dos punts i el mode de transport pels que es vol calcular la ruta. En canvi, els paràmetres de sortida descriuen l'estructura que seguiran les dades. Per exemple, les dades d'aparcament de Les Corts, oferides per Worldsensing, tindran els següents paràmetres de sortida: latitud, longitud i disponibilitat del lloc d'aparcament.

També es pot observar que les recomanacions tenen nom, descripció i àrea d'afectació, aquesta última característica representada per un polígon (conjunt de localitza-

cions).

En canvi, una alarma és l'encarregada de monitorar dades d'aparcament o de trànsit. Aquest monitoratge permet identificar canvis a les dades monitorades al llarg del temps, activant o desactivant recomanacions associades a l'alarma en funció de l'estat d'aquestes dades.

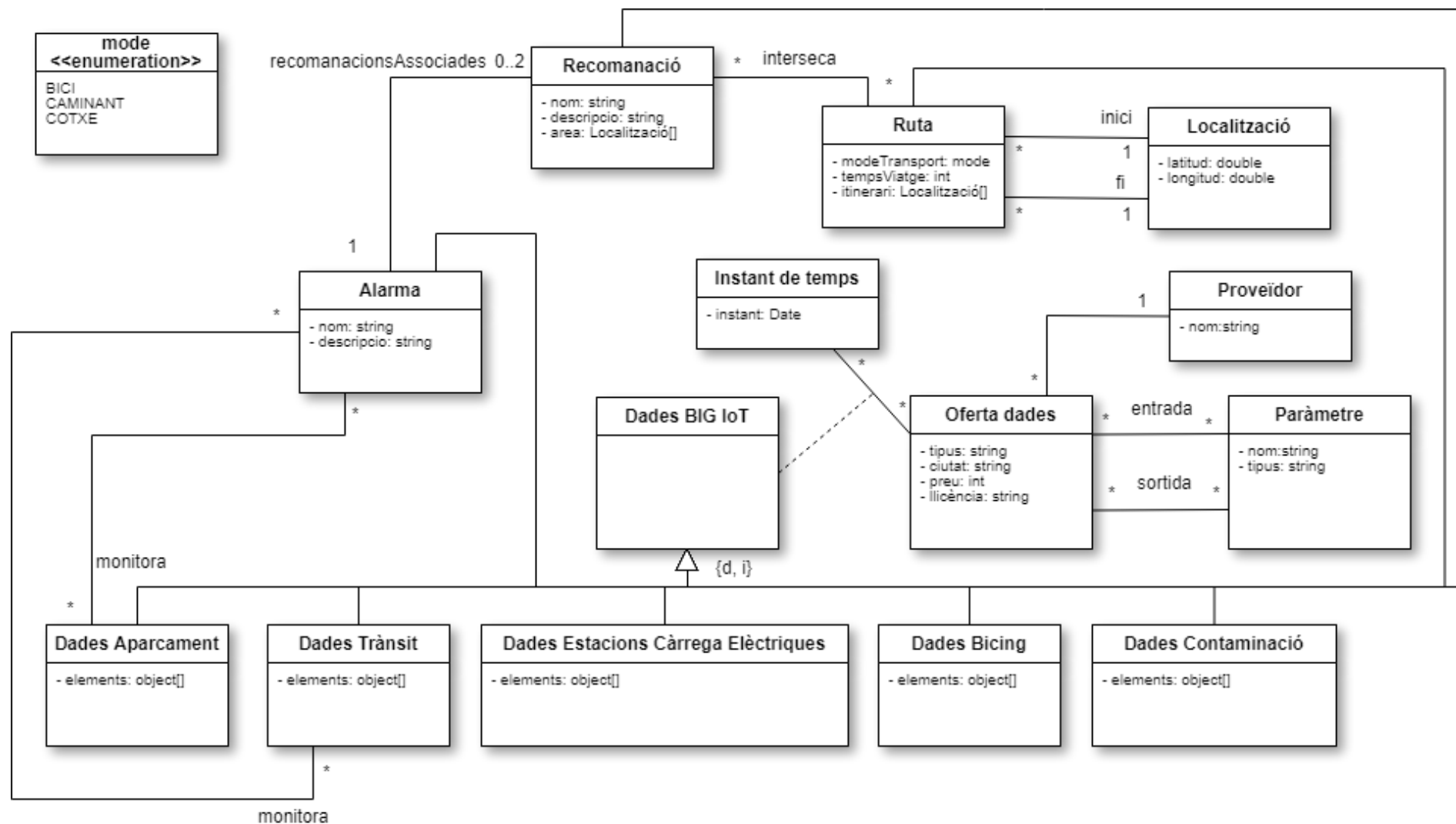


Figura 4: Model conceptual

9 Disseny

En aquest apartat es detalla l'arquitectura general i, concretament, la del servei d'alarmes. També s'expliquen les tecnologies i patrons emprats durant el desenvolupament. Finalment, es presenten els diagrames de classes de les aplicacions web i mòbil.

9.1 Arquitectura

La Figura 5 mostra els components que s'han desenvolupat en aquest treball. El servidor de producció (GIBO) és una màquina Ubuntu que desplega els serveis i aplicacions desenvolupats utilitzant tecnologies de virtualització Docker. El desplegament és detallat a la secció Desplegament, apartat 11.

Cada servei i aplicació està connectat al *Marketplace* de manera homogènia, tots utilitzen la llibreria BIG IoT java-lib. Els serveis d'aparcament i trànsit retornen les dades obtingudes de l'*API* de Worldsensing. Els serveis d'estacions de càrrega elèctrica i Bicing retornen les dades obtingudes de la plataforma OpenData BCN. El servei encarregat de retornar una ruta entre dos punts (*Routing Service*) utilitza un proveïdor extern de rutes, proporcionat pels col·laboradors BIG IoT de la regió de Piedmont, Itàlia. Per tant, podem dir que aquests serveis BIG IoT actuen com a passarel·la entre les plataformes proveïdores de dades i l'ecosistema BIG IoT (*Marketplace*).

En el diagrama no es mostren les interaccions entre components ja que el *Marketplace* és l'eix central de l'arquitectura que possibilita i facilita la comunicació entre qualssevol component que hi estigui connectat.

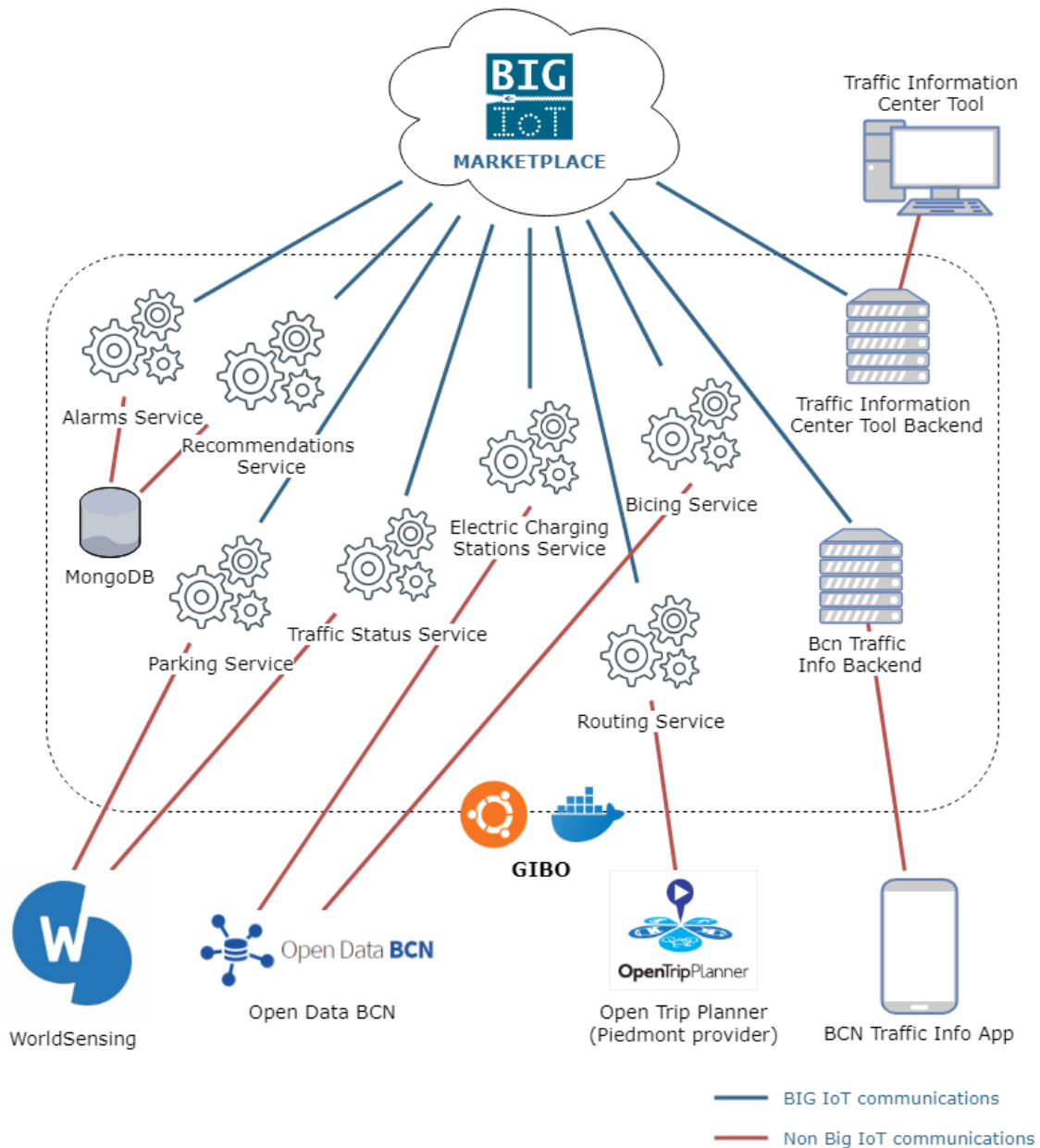


Figura 5: Arquitectura general

9.1.1 Servei d'alarmes

És important detallar l'arquitectura i funcionament del sistema d'alarmes, ja que integra el sistema de monitoratge que afegeix complexitat a l'arquitectura.

La Figura 6 mostra els diferents mòduls que conformen el servei d'alarmes. El *BIG IoT Service Module* és l'encarregat d'oferir les funcionalitats del servei d'alarmes a l'ecosistema BIG IoT, al *Marketplace*. Aquest mòdul és també l'encarregat de gestionar la persistència de les alarmes, emmagatzemades a una base de dades MongoDB.

El sistema de monitoratge s'estructura en dos mòduls: *Query Engine* i el *Monitor Module*. *Query Engine* és el mòdul desenvolupat pel Dr. Xavier Oriol que permet definir i processar consultes a alt nivell, en llenguatge *Datalog*, de les dades BIG IoT. Aquest mòdul ha estat extret d'un altre projecte del grup d'investigació MPI (Modelització i processament de la informació) de la UPC [18].

Per exemple, aquest mòdul permet realitzar una consulta que retorni tots els trams de les Rondes en que la velocitat actual sigui menor a 50 Km/h. És a dir, aquest mòdul accepta consultes semblants a consultes SQL, però es té en compte la semàntica BIG IoT que determina quins són els paràmetres clau a monitorar. Per tal de funcionar correctament, aquest mòdul necessita accedir a una base de dades SQL.

Per altra banda, el *Monitor Module* és el mòdul encarregat de, per a cada alarma existent al sistema, realitzar la consulta corresponent al *Query Engine* i actualitzar el conjunt de dades que està sent monitorat. Atès que aquest sistema monitora les dades mitjançant consultes, aquestes poden ser construïdes de tal manera que es considera activa aquella alarma que té elements retornats pel *Query Module*, i inactiva en cas contrari.

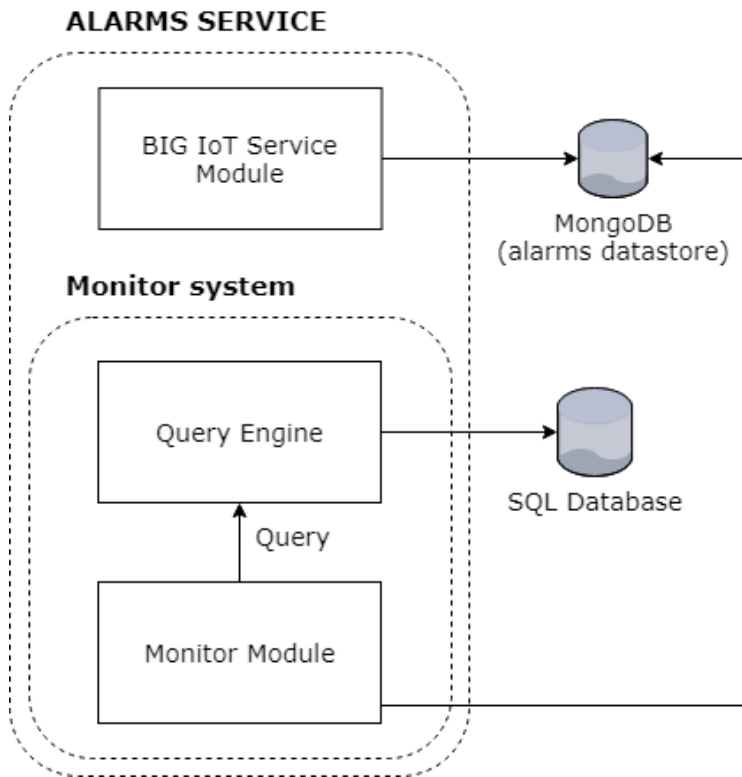


Figura 6: Arquitectura del servei d'alarmes

9.2 Tecnologia

En aquest apartat es detallen les diferents tecnologies emprades durant el desenvolupament d'aquest treball.

Java

Java és un llenguatge de programació orientat a objectes altament utilitzat, amb una sòlida comunitat d'usuaris i extensa documentació. S'ha escollit aquest llenguatge per programar els serveis BIG IoT i els *backends* de les aplicacions web i mòbil perquè la llibreria proporcionada per BIG IoT és en Java.

Typescript + HTML/CSS

Typescript és un llenguatge de programació basat en Javascript que afegeix tipus estàtics i objectes basats en classes. Juntament amb HTML i CSS aquesta tríada de llenguatges ha esdevingut el principal recurs per al desenvolupament d'aplicacions web. S'ha escollit aquest llenguatge pel seu extens ús, la gran comunitat de programadors que l'utilitza i l'abundant documentació que es pot trobar. També són els llenguatges utilitzats pels frameworks descrits a continuació.

Angular.io

Angular.io és un framework de desenvolupament d'aplicacions web que utilitza Typescript, i HTML/CSS, com a llenguatges de programació. L'objectiu principal d'aquest framework és facilitar el desenvolupament i manteniment d'aplicacions web, proporcionant estructura i recursos basats en el patró Model Vista Controlador (MVC).

Spring

Spring és un framework de desenvolupament d'aplicacions en Java. Aquest framework està estructurat en extensions i mòduls que poden ser instal·lats segons els requisits de l'aplicació a desenvolupar. Aquest framework també utilitza el patró de disseny Model Vista Controlador (MVC) de base. En aquest treball s'utilitzarà principalment el mòdul SpringBoot que facilita el desenvolupament de *REST API*'s. És el framework utilitzat per desenvolupar el backend de les aplicacions web i mòbil.

Ionic

Ionic és un framework basat en Angular que proporciona eines i serveis que faciliten el desenvolupament d'aplicacions híbrides mitjançant tecnologies web com HTML/CSS. Aquest és el framework escollit per desenvolupar l'aplicació per a mòbils *BCN Traffic Info*.

JHipster

JHipster és un framework que utilitza Angular pel *frontend* i Spring per al *backend*. Proporciona una aplicació web inicial amb un sistema de gestió d'usuaris i intercanvi

de vistes pre-dissenyat. Aquest és el framework escollit per desenvolupar l'aplicació web *Traffic Information Center Tool*.

Bootstrap

Bootstrap és un framework que facilita el desenvolupament en CSS. Proporciona eines per editar fàcilment l'aparença i posicionament dels components diferents web. Aquest framework està integrat tant a Angular.io com a Ionic, de manera que s'utilitza pel desenvolupament de les dues aplicacions (web i mòbil).

MongoDB

MongoDB és un sistema de base de dades NoSQL orientat a documents, desenvolupat mitjançant codi obert. És utilitzat per gestionar la persistència de l'aplicació web *Traffic Information Center Tool* i els sistemes d'alarmes i recomanacions.

9.3 Patrons

Per al desenvolupament del projecte s'han utilitzat els següents patrons:

Patró factoria + *Singleton*

Aquest és un patró altament utilitzat en Java. Permet modularitzar el codi, la classe factoria és l'encarregada de retornar les instàncies d'una classe determinada. És altament útil quan aquesta classe és heredada per altres classes o per quan es vol mantenir un control del número d'instàncies de la classe present al sistema. La instància de la classe factoria és única al sistema (*Singleton*). Aquest patró s'utilitza per gestionar els Services que proporcionen dades en temps real, al Backend de l'aplicació web *Traffic Information Center Tool*.

Patró *Database Access Object (DAO)*

Aquest patró permet una comunicació homogènia entre les aplicacions i la base de dades mitjançant una *interface*. L'adopció d'aquest patró garanteix una reducció de l'acoblament entre les aplicacions i la base de dades. Concretament s'ha implementat als serveis de recomanacions i alarmes, que era necessari emmagatzemar recomanacions i alarmes a una base de dades No-Sql, *mongoDB*.

Model Vista Controlador (MVC)

Aquest patró separa en diferents components les dades, la lògica de negoci i la vista. Millora el manteniment del codi i és àmpliament utilitzat pels *frameworks* més comuns. Aquest és el patró utilitzat per *Angular.io*, *framework* amb els que es desenvolupa l'aplicació web *Traffic Information Center Tool* i l'aplicació mòbil *BCN Traffic Info*.

9.4 Diagrames de classes

En aquest apartat es detallen els diagrames de classes de l'aplicació web *Traffic Information Center Tool* i l'aplicació mòbil *BCN Traffic Info*.

9.4.1 *Traffic Information Center Tool*

A continuació es mostra el diagrama de classes, *frontend* i *backend*, de l'aplicació web *Traffic Information Center Tool*. Cal recordar que aquesta aplicació web ha estat desenvolupada mitjançant el *framework JHipster* que proporciona una plantilla amb funcionalitats bàsiques com la gestió d'usuaris o la navegabilitat entre vistes. Els següents diagrames només es detallen les classes implementades amb els corresponents mètodes, rellevants per entendre el funcionament del sistema.

Backend

La Figura 7 mostra el diagrama de les classes implementades, utilitzant el *framework Spring.io*. L'anotació *resource* indica que la classe conté mètodes que s'exposen mitjançant una url determinada, són la via de comunicació entre el *frontend* i *backend*. També es fa ús dels *services*, classes que s'executen en paral·lel proporcionant *sockets* que el *frontend* utilitza per obtenir les dades en accés continu (*live*).

Frontend

La Figura 8 mostra el diagrama de les classes implementades, utilitzant el *framework Angular.io*. En aquest cas les classes anotades amb *Component* són les encarregades de mostrar les diferents vistes. Una mateixa vista pot estar dividida en diversos *Components*. En aquest cas les classes *Service* proveeixen diferents mètodes i funcions a la resta de *Components*. Les classes *Service* existents en aquest sistema s'encarreguen de la comunicació amb el *backend* (accés únic i accés continu mitjançant *sockets*) i de la construcció dels marcadors mostrats al mapa.

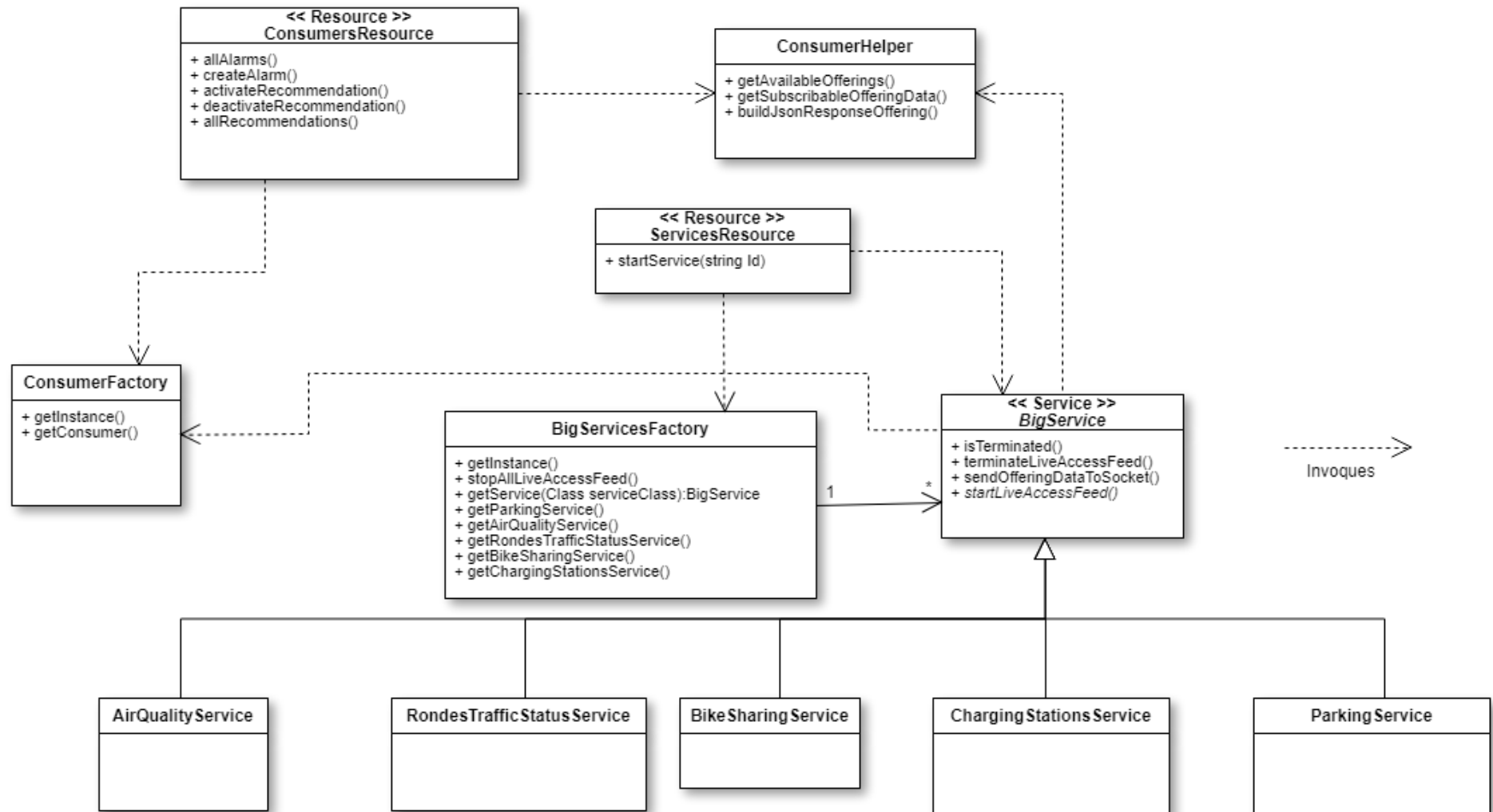


Figura 7: Diagrama de classes del *backend* del *Traffic Information Center Tool*

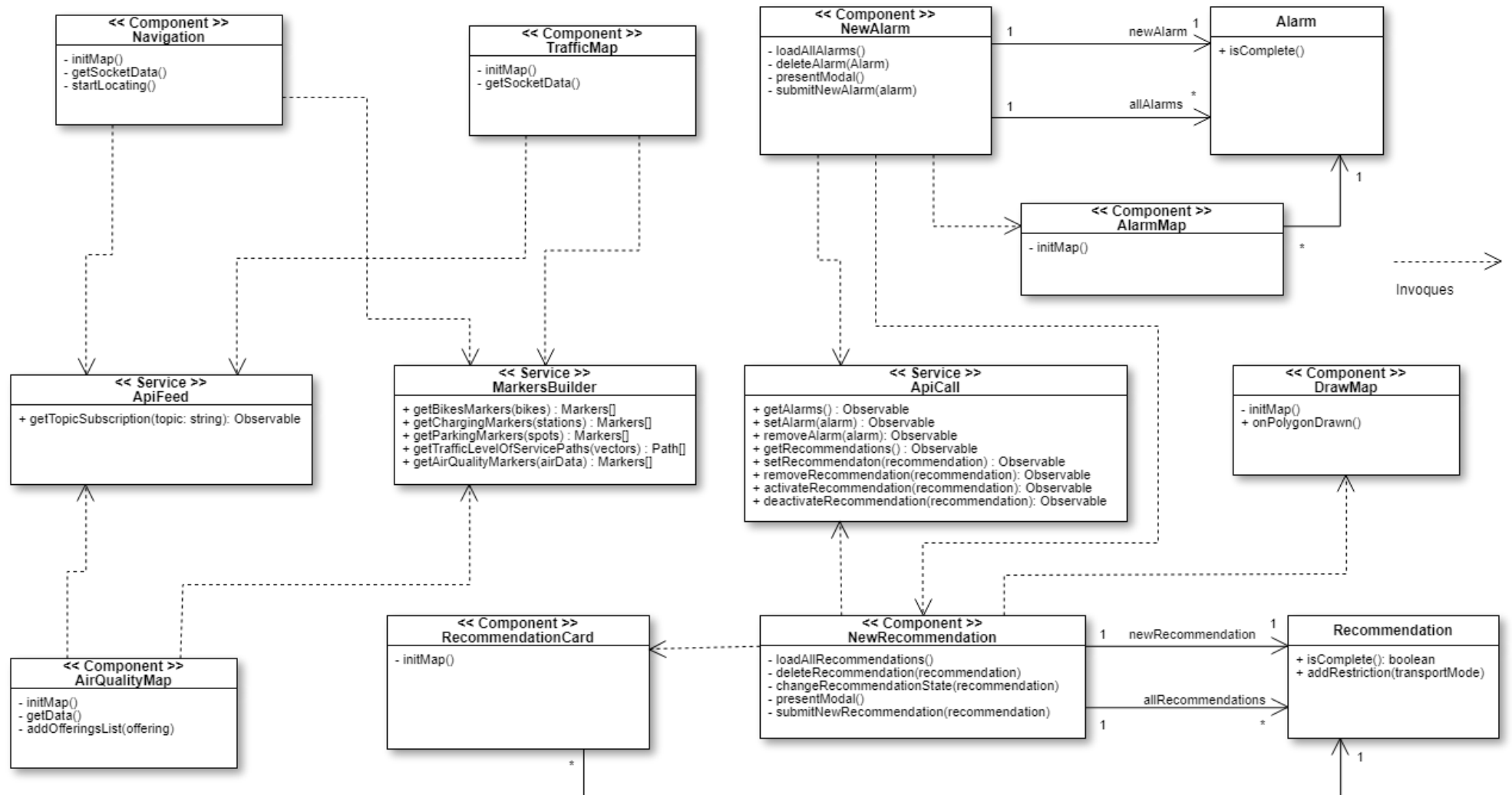


Figura 8: Diagrama de classes del *frontend* del *Traffic Information Center Tool*

9.4.2 *BCN Traffic Info*

A continuació es mostra el diagrama de classes, *frontend* i *backend*, de l'aplicació mòbil *BCN Traffic Info*. Cal recordar que aquesta aplicació ha estat desenvolupada mitjançant els *frameworks* *Ionic* i *Spring.io* (*frontend* i *backend* respectivament) que proporcionen plantilles base. Els següents diagrames detallen les classes implementades amb els corresponents mètodes, rellevants per entendre el negoci del sistema.

Backend

La Figura 9 mostra el diagrama de les classes implementades, utilitzant el *framework* *Spring.io*. L'anotació *resource* indica que la classe conté mètodes que s'exposen mitjançant una url determinada, són la via de comunicació entre el *frontend* i *backend*.

Frontend

La Figura 10 mostra el diagrama de les classes implementades, utilitzant el *framework* *Ionic* basat en *Angular.io*. En aquest cas les classes anotades amb *Component* i *Page* són les encarregades de mostrar les diferents vistes. Una mateixa vista pot estar dividida en diversos *Components* i *Pages*. Una *Page* és un tipus de *Component*, generalment és el *Component* principal que conté la resta *Components* en una mateixa vista. La classe *Service* és l'encarregada de la comunicació amb el *backend*.

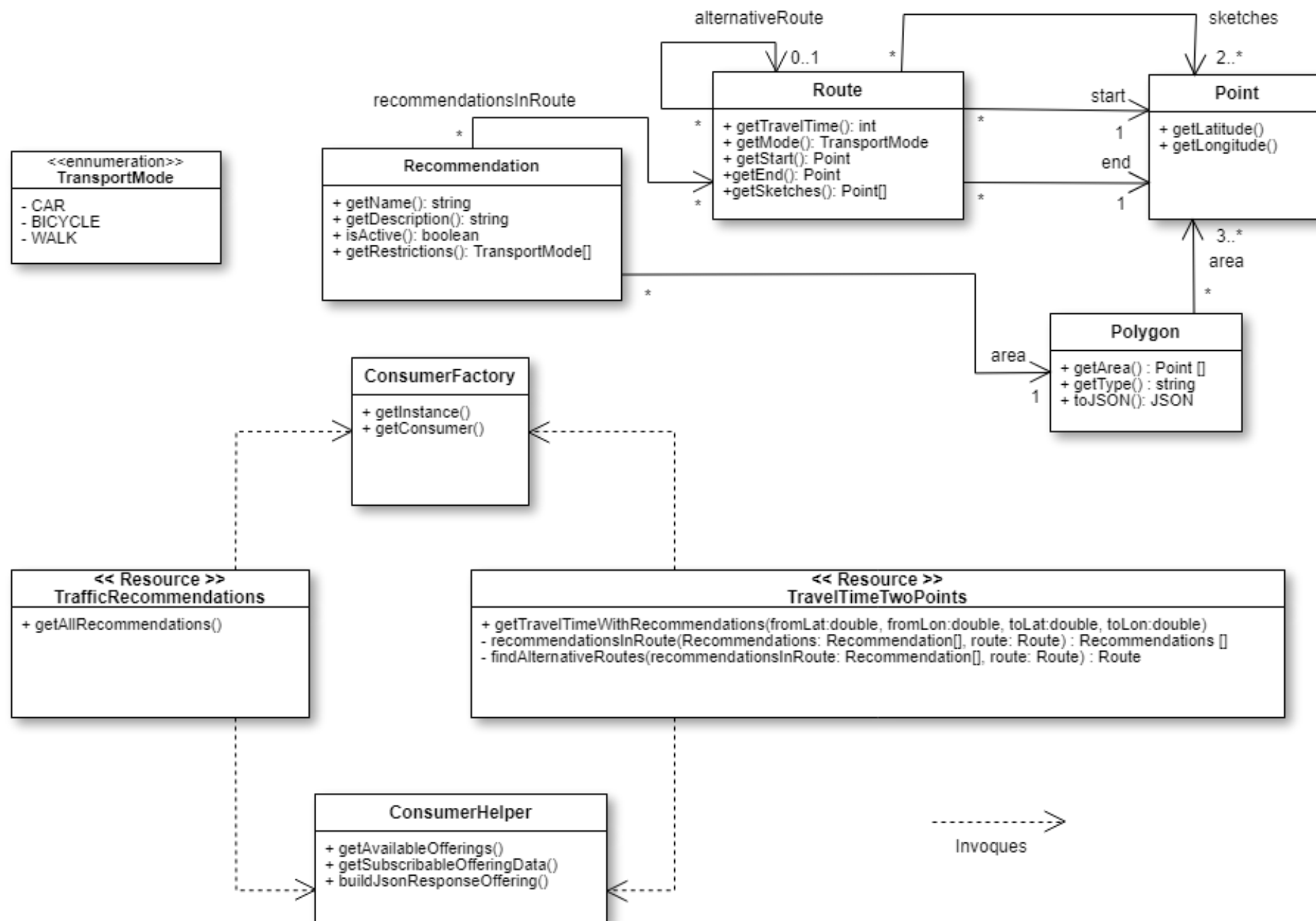


Figura 9: Diagrama de classes del *backend* de la *BCN Traffic Info*

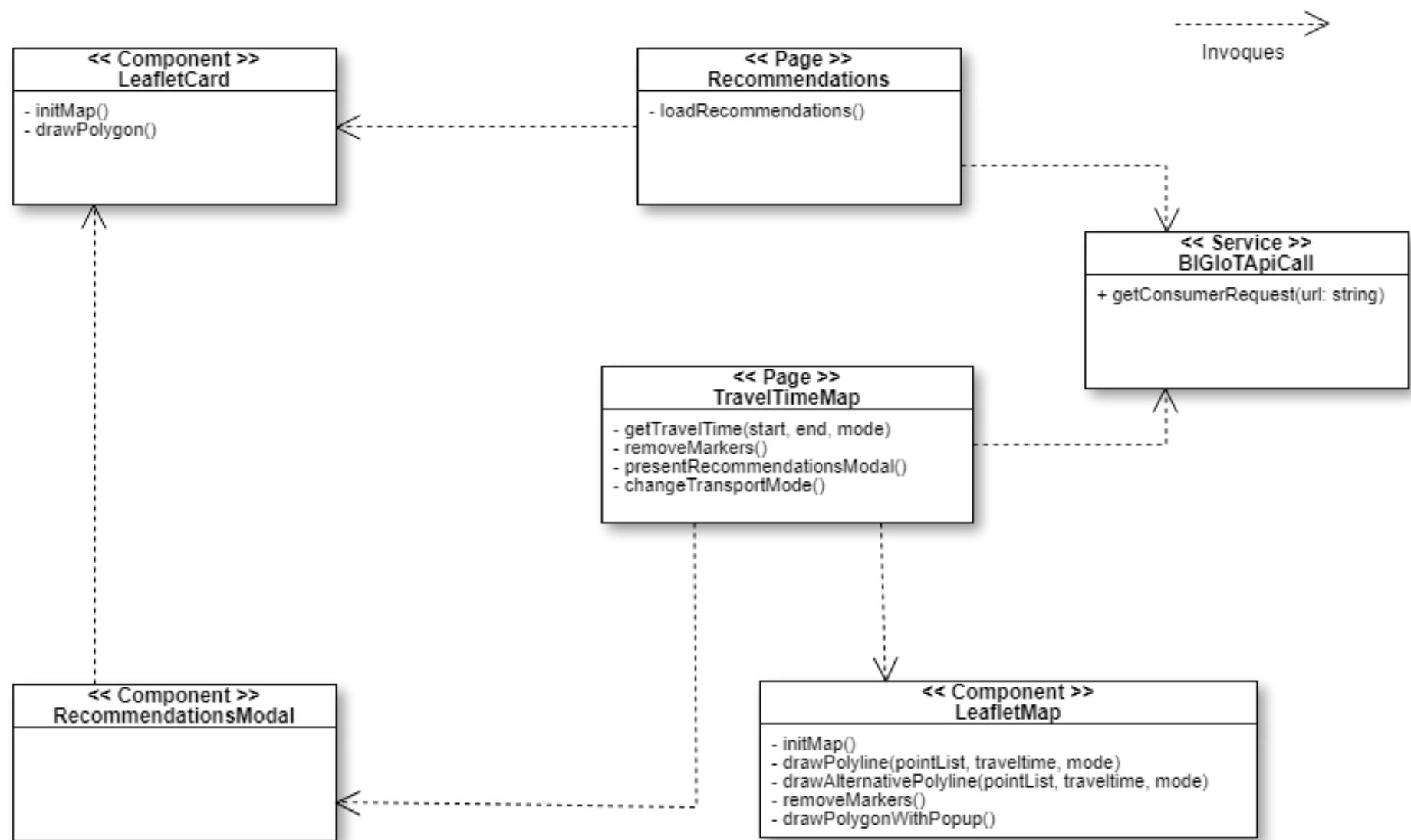


Figura 10: Diagrama de classes del *frontend* de la *BCN Traffic Info*

10 Implementació

10.1 Tasques a realitzar

En aquest treball s'han desenvolupat tasques de diferents components, per aquest motiu s'ha decidit organitzar-les en paquets de treball. Aquests paquets de treball contenen les tasques d'un mateix component o similars. A continuació es detallen els diferents paquets de treball.

Paquet de treball 1

Paquet que conté les tasques de la fase inicial del projecte. S'analitza l'estat de l'art, es prepara l'organització de l'equip i els entorns de desenvolupament i producció.

- Definir requisits
- Estat de l'art
- Organització equip
- Disseny general
- Eines treball i entorn desenvolupament
- Preparació entorn producció

Paquet de treball 2

En aquest paquet de treball s'introdueix el projecte BIG IoT i es desenvolupen les primeres proves de concepte amb les tecnologies proporcionades per BIG IoT: *Marketplace*, llibreria i capa semàntica.

- Planificació
- Proves concepte llibreria BIG IoT
- Proves concepte desplegament continu

Paquet de treball 3

En aquest paquet de treball s'agrupen les tasques relacionades amb el desenvolupament de serveis BIG IoT. Primerament s'analitzen les plataformes que proporcionen

les dades. Els serveis desenvolupats en aquest paquet de treball oferiran aquestes dades al *Marketplace*.

- Planificació
- Proves concepte fonts de dades
- Servei dades bicing
- Servei estacions de càrrega elèctrica
- Servei enrutament
- Integració serveis desenvolupats per Worldsensing

Paquet de treball 4

Aquest paquet de treball conté les tasques relacionades amb el panell de control, el *Traffic Information Center Tool*. A l'inici s'analitzen les tecnologies a utilitzar i després es desenvolupa el *dashboard*, integrant els serveis desenvolupats al paquet de treball anterior.

- Planificació
- Proves concepte framework Jhipster
- Backend: Obtenció dades, mètode accés únic
- Backend: Obtenció dades, mètode accés continu
- Frontend: Dashboard
- Frontend: Alarmes i recomanacions

Paquet de treball 5

Aquest paquet de treball conté les tasques relacionades amb l'aplicació per a mòbil, la *Bcn Traffic Info*. Integra també serveis desenvolupats al paquet de treball 3.

- Planificació
- Proves concepte framework Ionic
- Backend: Extern
- Comunicació backend
- Integració servei enrutament

- Integració recomanacions
- Integració routing amb recomanacions

Paquet de treball 6

Aquest paquet de treball es desenvolupa paral·lelament amb els paquets de treball finals, tal i com es mostra al diagrama de Gantt. Conté les tasques relacionades amb la documentació i preparació de demostracions.

- Documentació
- Preparació demostracions
- Release final

10.2 Exemples de codi BIG IoT

La llibreria java-lib proporcionada per BIG IoT permet la comunicació homogènia entre les plataformes proveïdors i consumidors de dades amb el *Marketplace*. Aquesta llibreria ofereix funcionalitats i classes que permeten:

- Autenticar-se al *Marketplace* com a proveïdor o consumidor de dades.
- Oferir dades mitjançant el *Marketplace*.
- Cerca i filtratge de dades.
- Obtenció de dades amb accés únic.
- Obtenció de dades amb accés continu.

El següent codi implementa l'oferiment de dades al *Marketplace* per part del proveïdor de dades Bicing Barcelona:

```
ProviderSpark provider = new ProviderSpark(PROVIDER_ID,
    MARKETPLACE_URI, PROVIDER_DNS_NAME, PROVIDER_PORT);

provider.authenticate(PROVIDER_SECRET);

// OFFERING CREATION
RegistrableOfferingDescription offeringDescription = provider.
    createOfferingDescription("BikesOffering")
        .WithName("BicingBikeSharing")
        .withCategory("urn:big-iot:BikeSharingStationCategory")

        .addOutputData("longitude", "http://schema.org/longitude",
            BigIotTypes.ValueType.NUMBER)
        .addOutputData("latitude", "http://schema.org/latitude",
            BigIotTypes.ValueType.NUMBER)
        .addOutputData("type", "http://schema.big-iot.org/mobility/
            bikeTypes", BigIotTypes.ValueType.TEXT)
        .addOutputData("status", "http://schema.big-iot.org/mobility/
            bikeSharingStationStatus", BigIotTypes.ValueType.TEXT)
        .addOutputData("slots", "http://schema.big-iot.org/mobility/
            numberOfAvailableBikeParkingSlots", BigIotTypes.ValueType.
            NUMBER)
        .addOutputData("bikes", "http://schema.big-iot.org/mobility/
            numberOfAvailableBikes", BigIotTypes.ValueType.NUMBER)

        .inRegion("Barcelona")
        .withPricingModel(BigIotTypes.PricingModel.FREE)
        .withLicenseType(BigIotTypes.LicenseType.OPEN_DATA_LICENSE)
        .withRoute("bikes") //implies mode 2 or mode 1
        .asHttpGet()
```

```

        .producesJson()
        .withAccessRequestHandler(accessCallback);

// OFFERING REGISTRATION
RegisteredOffering offering = offeringDescription.register();

```

Primerament s'instancia la classe **Provider** que permet l'autenticació amb el *Marketplace* utilitzant l'identificador i secret del proveïdor (prèviament obtinguts mitjançant el la pàgina web del *Marketplace*). Seguidament, es descriu l'oferiment de dades (*offering*) instanciant la classe **RegistrableOfferingDescription**. En aquest punt s'especifiquen les principals característiques de l'*offering*: nom, categoria, regió, preu, llicència i format de retorn. També es detallen els paràmetres de sortida, és a dir, quin serà el contingut i format de les dades que s'ofereixen.

En aquest punt cal recordar que BIG IoT proporciona una capa semàntica per tal de descriure les dades. *Schema.org* és un projecte impulsat per Google, Microsoft, Yahoo and Yandex, mantingut per la comunitat amb l'objectiu de proporcionar esquemes que permetin descriure dades i entitats del món que ens envolta. BIG IoT crea una extensió d'aquest projecte per tal d'afegir-hi les descripcions semàntiques relacionades amb els casos d'ús utilitzats a les proves pilot BIG IoT.

La operació **addOutputData(String, RDFType, ValueType)** permet la descripció de cada paràmetre; el primer argument de la funció determina el nom del paràmetre, el segon determina la descripció semàntica del paràmetre i el tercer el tipus. En aquest cas, les dades retornades tindran una geolocalització (latitud i longitud), un atribut type, un atribut status, un atribut slots i un atribut bikes. L'*url* que s'utilitza per identificar el RDFType, al accedir-hi, també proporciona tota la informació existent de l'atribut.

A l'oferiment de dades també s'hi vincula una funció (**accessCallback**) mitjançant el mètode **withAccessRequestHandler(AccessRequestHandler)**. L'argument d'aquest mètode és la funció que serà executada cada vegada que es demanin les dades d'aquest *offering*, és per tant, la funció que determina el negoci d'aquest *offering*.

Per últim, l'*offering* és registrat al *Marketplace* mitjançant la funció **register()**.

El següent codi mostra el funcionament de la funció, **accessCallback**, encarregada de retornar les dades de l'*offering* de cada petició.

```

private static AccessRequestHandler accessCallback = new
    AccessRequestHandler() {
        @Override
        public BigIotHttpResponse processRequestHandler(
            OfferingDescription offeringDescription,
            Map<String, Object> inputData, String subscriptionId, String
            consumerInfo) {

```



```
// Obtencio de les dades
return BigIotHttpResponse.okay().withBody(stations.toString()).
    asJsonType();
}
```

El següent codi exemplifica la cerca de dades, *discovery*.

```
// Construct Offering search query incrementally/
OfferingQuery query = OfferingQuery.create("DemoParkingQuery")
    .withName("Demo_Parking_Query")
    .withCategory("urn:big-iot:ParkingSpaceCategory")
    .inCity("Barcelona")
    .addInputData(new RDFType("schema:longitude"), ValueType.NUMBER)
    .addInputData(new RDFType("schema:latitude"), ValueType.NUMBER)
    .addOutputData(new RDFType("schema:longitude"), ValueType.NUMBER)
    .addOutputData(new RDFType("schema:latitude"), ValueType.NUMBER)
    .addOutputData(new RDFType("datex:parkingSpaceStatus"),
        ValueType.TEXT)
    .withPricingModel(BigIotTypes.PricingModel.PER_ACCESS)
    .withMaxPrice(Euros.amount(0.5))
    .withLicenseType(LicenseType.CREATIVE_COMMONS);

// Discover available offerings based on Offering Query
List<SubscribableOfferingDescription> list = consumer.discover(query).
    get();
```

Mitjançant la classe **OfferingQuery** es construeix l'objecte **query** que determina les característiques de les ofertes de dades que s'estan cercant. En aquest cas, es cerquen ofertes de dades d'aparcament de la ciutat de Barcelona que acceptin com a paràmetres d'entrada una posició determinada per una latitud i una longitud. L'oferta de dades ha de retornar les dades amb l'estructura: latitud, longitud i estat de la plaça d'aparcament. La **query** també filtra les ofertes de dades per preu i llicència.

Un cop s'ha escollit l'oferta de dades desitjada, l'accés a les dades es pot realitzar de dues maneres diferents; accés únic o accés continu.

```
// PREPARE ACCESS PARAMETERS
AccessParameters accessParameters = AccessParameters.create()
    .addRdfTypeValue("schema:latitude", 42.0)
    .addRdfTypeValue("schema:longitude", 9.0);
```

```
// ACCESS ONE TIME
```

```
CompletableFuture<AccessResponse> response = offering.accessOneTime(
    accessParameters);

// CONTINUOUS ACCESS
AccessFeed accessFeed = offering.accessContinuous(accessParameters,
    feedDuration.getMillis(),
    feedInterval.getMillis(),
    (f, r) -> logger.info("Incoming_feed_data:_" + r.asJsonNode().
        size() + "_elements_received."),
    (f, r) -> logger.info("Feed_operation_failed"));
```

Mitjançant la classe **AccessParameters** es determinen els paràmetres d'accés a l'oferta de dades. La instrucció que permet l'accés únic retorna les dades estructurades a la classe **AccessResponse**. En canvi, l'accés continu es conforma adjuntant-hi la funció que s'executarà a cada interval de l'accés.

11 Desplegament

Els diferents components d'aquest treball s'han desplegat mitjançant la tecnologia Docker. Docker és una eina de virtualització que permet crear l'entorn adequat a les dependències que requereix l'artefacte que s'haurà d'executar, aquest entorn personalitzat s'anomena imatge. Les imatges són independents entre sí i comparteixen recursos comuns de la màquina que les allotja, màquina hoste, mitjançant un sistema de virtualització basat en capes. Les imatges són executades pel que s'anomenen contenidors. Aquest sistema de desplegament permet generar entorns d'execució diferents i independents en una mateixa màquina, compartint els recursos comuns.

11.1 Servidor de producció i eines de monitoratge

Cada servei BIG IoT és desplegat a una imatge docker, i cada una d'aquestes és executada per un contenidor. L'aplicació web i el *backend* de l'aplicació mòbil també són desplegats en imatges i contenidors docker independents.

Per tal de monitorar el servidor de producció i els contenidors en execució, s'ha utilitzat un sistema anomenat Prometheus, que permet obtenir i integrar informació sobre els recursos de la màquina hoste.

La informació bàsica dels recursos de la màquina és obtinguda directament per Prometheus. De totes maneres, Prometheus també pot obtenir dades de recursos més específics mitjançant mòduls independents especialitzats. Per exemple, el mòdul Cadvisor proporciona informació sobre els recursos de cada contenidor Docker en execució.

Entre d'altres funcionalitats Prometheus proporciona un llenguatge per tal de realitzar consultes, més o menys complexes, o crear una alarma que monitori un cert recurs de la màquina.

Aquesta informació és mostrada a un panell de control desenvolupat amb Graphana.

A la Figura 11 es mostra el panell de monitoratge del servidor. A la part superior es poden apreciar estadístiques pròpies de la màquina: CPU, memòria RAM, SWAP i quota de disc. A la part inferior es mostren els històrics del consum de recursos dels contenidors Dockers: CPU, memòria RAM i xarxa.

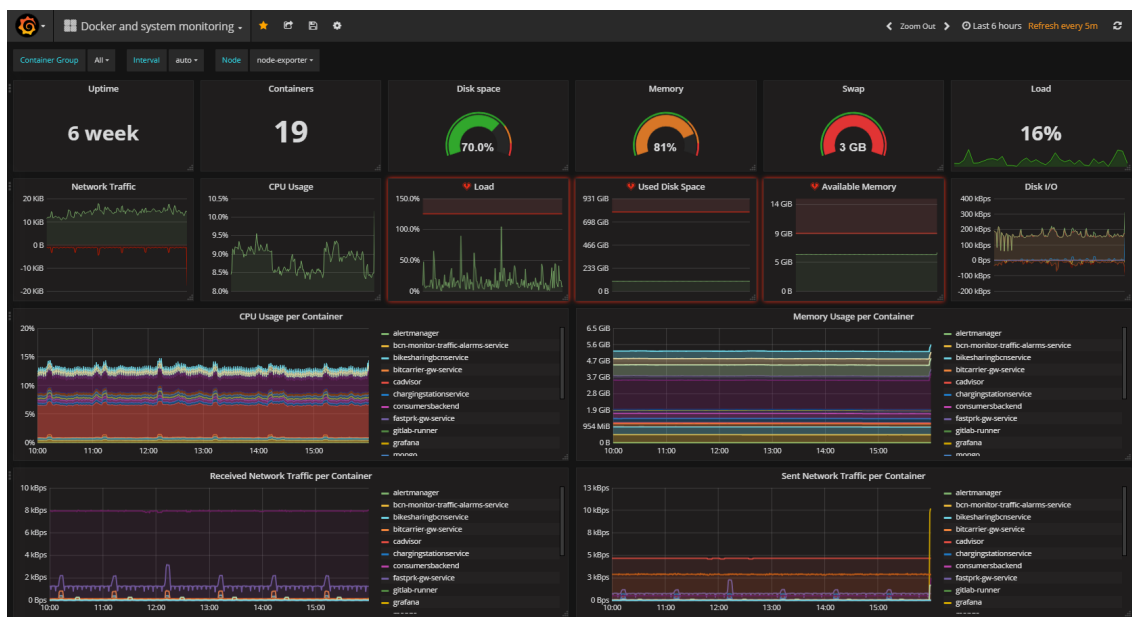


Figura 11: Panell de monitoratge Graphana

Els diferents contenidors i imatges Docker es gestionen mitjançant línia de comandes. De totes maneres, Docker proporciona una interfície gràfica, Portainer.io, que facilita la gestió Docker de la màquina. Entre d'altres funcionalitats més avançades, Portainer.io permet veure el llistat dels contenidors i imatges Docker existents a la màquina. També permet realitzar accions bàsiques com engegar un contenidor, parar-lo o posar-lo en pausa. També permet visualitzar les configuracions de cada imatge i contenidor. La figura 12 en mostra una captura.

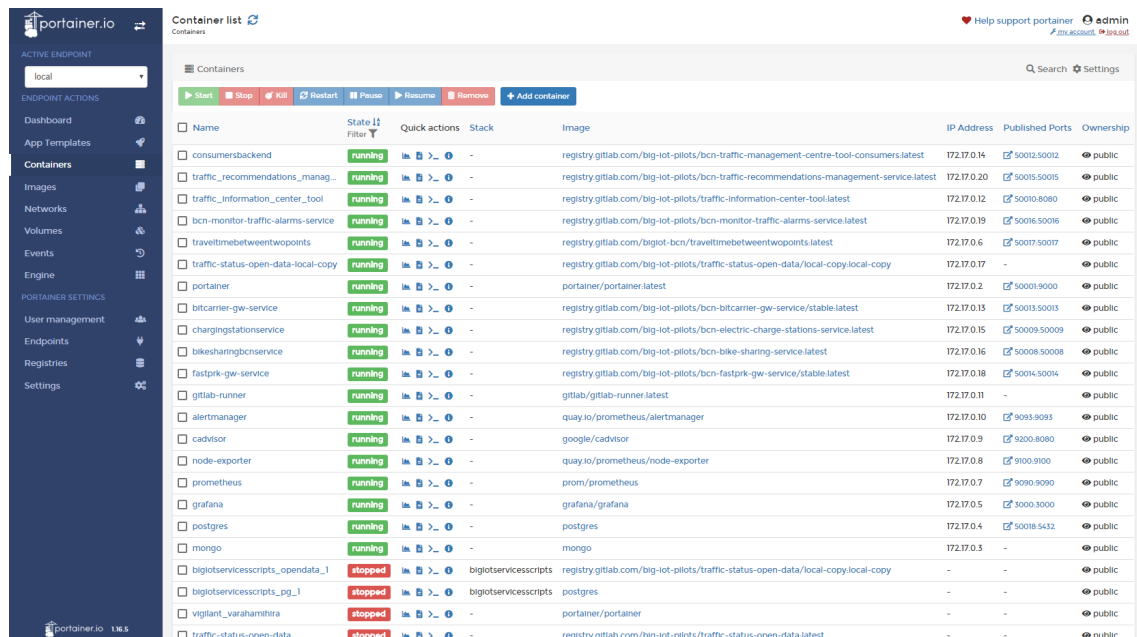


Figura 12: Panell de gestió Docker Portainer

11.2 Integració contínua

La integració contínua és una metodologia que consisteix en compilar i realitzar proves automàticament a mesura que avança el desenvolupament del software, detectant així errades al codi el més aviat possible.

Gitlab és el repositori de codi utilitzat en aquest treball. Gitlab proporciona als usuaris l'eina d'integració contínua anomenada Gitlab-CI; permet configurar un entorn a cada repositori, adequat a les dependències del codi. En aquest entorn es poden executar diversos cicles, a cada un d'ells es poden dur a terme diverses tasques com executar testos, compilar el codi, construir artefactes per producció, etc. La configuració dels entorns i cicles és determinada pel fitxer `.gitlab-ci.yml` situat a l'arrel del projecte.

Gitlab-CI també facilita el desplegament amb tecnologies Docker. De fet, proporciona un registre per a cada repositori en que s'emmagatzemen les imatges Docker construïdes durant els cicles d'integració contínua.

La construcció i definició dels cicles i entorn és semblant a la tecnologia Docker; es va construint l'entorn per capes partint d'un sistema operatiu base (imatge base). La figura 13 mostra la configuració Gitlab-CI de l'aplicació web *Traffic Information Center Tool* que il·lustra la creació de l'entorn i cicles requerits.

```
1  image: docker:latest
2
3  stages:
4    - build
5    - release
6
7  before_script:
8    - chmod +x gradlew
9
10 build-develop:
11   image: java:8-jdk-alpine
12   stage: build
13   script:
14     - apk add --update nodejs
15     - apk add --update yarn
16     - ./gradlew -Pprod clean bootRepackage
17   only:
18     - develop
19
20 release-build-docker:
21   image: docker:latest
22   services:
23     - docker:dind
24   stage: release
25   script:
26     - apk --update add openjdk8
27     - apk add --update nodejs
28     - apk add --update yarn
29     - ./gradlew bootRepackage -Pprod buildDocker
30     - docker login -u gitlab-ci-token -p $CI_BUILD_TOKEN registry.gitlab.com
31     - docker push registry.gitlab.com/big-iot-pilots/traffic-information-center-tool
32   only:
33     - master
```

Figura 13: Configuració Gitlab-CI *Traffic Information Center Tool*

En aquest cas, s'executen dos cicles (*stages*), build-develop i release-build-docker, aquest últim només a la branca master que conté el codi preparat per ser executat a producció. A l'*stage* build-develop s'utilitza com a imatge base un sistema operatiu linux Alpine amb java 8 instal·lat (línia 11). S'instal·len també les dependències necessàries per compilar el codi: nodejs i yarn (línies 14 i 15). Finalment, es compila i executen els tests mitjançant la instrucció `./gradlew -Pprod clean bootRepackage` (línia 16).

L'*stage* release-build-docker utilitza com a imatge base un sistema operatiu linux

Alpine amb Docker instal·lat. En aquest cas també s'instal·len les dependències necessàries per executar el codi: java 8, nodejs i yarn. En aquest cas, l'objectiu d'aquesta *stage* és construir una imatge Docker del software preparada per ser executada a producció. Per tant, es construeix la imatge Docker mitjançant la instrucció `./gradlew bootRepackage -Pprod buildDocker`. Finalment es transfereix la imatge creada al registre d'imatges del repositori mitjançant les instruccions `docker login` i `docker push` que es poden apreciar a la imatge.

12 Validacions

En aquest apartat es detallen els dos tipus de validacions existents per aquest treball final de grau.

12.1 Validacions de les parts interessades

En aquest treball es plantegen tres nivells de validacions extretes del *feedback* provinent de les diverses parts interessades.

Les validacions de primer nivell s'extreuen a partir de les reunions que es realitzen a final d'Sprint amb els membres de l'equip de desenvolupament. Durant aquestes reunions es revisen les tasques realitzades al llarg de l'últim Sprint i, mitjançant petites demostracions dels components desenvolupats, es detecten errors i possibles millores. Aquest mètode iteratiu permet validar que els components s'estan desenvolupant d'acord amb els requisits i objectius establerts.

Durant el període de desenvolupament també es realitzen demostracions amb els col·laboradors del pilot de Barcelona. Aquestes demostracions serveixen per determinar si els artefactes desenvolupats fins el moment s'ajusten als objectius dels *stakeholders* del pilot i, en conseqüència, d'aquest treball.

L'últim nivell, i més important, és la validació a nivell europeu rebuda per part dels revisors de la Comunitat Europea. Són ells els que dictaminen si el projecte BIG IoT es desenvolupa correctament. D'aquí se'n deriva la correcta evolució dels pilots i, finalment, d'aquest treball final de grau.

Durant el desenvolupament d'aquest treball final de grau s'han realitzat dues demostracions a nivell europeu; a Atenes i Munich, en que es va demostrar la correcta interacció entre els serveis d'aparcament desenvolupats al pilot de Barcelona i el pilot del nord d'Alemanya.

12.2 *Testing*

El software desenvolupat també ha sigut validat mitjançant tests que comproven que el codi realitza les funcionalitats esperades.

En aquest treball s'ha desenvolupat un mòdul encarregat de testejar els diferents serveis BIG IoT implementats, mitjançant el *framework* de testing en Java JUnit. Aquest mòdul comprova que els serveis BIG IoT estan disponibles al *Marketplace* i que l'estructura de les dades ofertes correspon a les descripcions semàntiques adients.

A més, a BIG IoT existeix un paquet de treball encarregat de la integració entre els diferents pilots del projecte. Per tal de facilitar-ne la integració, aquest paquet de treball planteja diversos indicadors que determinen si els components software desenvolupats assoleixen un mínim de qualitat i estan preparats per a ser integrats amb la resta de components.

Els indicadors plantejats són els següents:

- **Marketplace correct connection (MC):** L'aplicació/servei es connecta correctament amb el *Marketplace*.
- **Marketplace correct offering discovery (MOD):** L'aplicació pot accedir correctament a les ofertes de dades desitjades.
- **Marketplace correct offering registry (MOR):** El servei ha registrat correctament l'oferta de dades al *Marketplace*.
- **Correct invocation of data offerings (IDO):** L'aplicació/servei obté les dades dels proveïdors mitjançant la llibreria BIG IoT.
- **Correct result retrieving (RR):** Les dades rebudes per l'aplicació/servei estan estructurades en format JSON vàlid.
- **Correct result offering (RO):** El servei retorna les dades en format JSON vàlid.
- **Correct data rendering (DR):** L'aplicació mostra correctament les dades obtingudes.

De totes maneres, BIG IoT finalitza l'any 2018 i està previst que durant la fase final del projecte es validi novament la correcta integració entre tots els components desenvolupats. Els indicadors descrits formen part d'una fase d'integració inicial. En un futur s'inclouran nous criteris de validació d'acord amb les necessitats del projecte BIG IoT. Per exemple, l'equip d'integració està treballant en un mòdul que automatitza les comprovacions anteriorment descrites, entre d'altres.

La Taula 1 mostra els indicadors en els components desenvolupats en aquest treball. En cas de que l'indicador no sigui adient pel component, es determina amb un *n/a*. Per exemple, els indicadors que fan referència a característiques d'un servei BIG IoT no són adients per a les aplicacions web i mòbil.

	MC	MOD	MOR	IDO	RR	DR	RO
Traffic Information Center Tool	sí	sí	n/a	sí	sí	sí	n/a
BCN Traffic Info	sí	sí	n/a	sí	sí	sí	n/a
Parking Service	sí	n/a	sí	n/a	n/a	n/a	sí
Alarms Service	sí	sí	sí	sí	sí	n/a	sí
Traffic Status Service	sí	n/a	sí	n/a	n/a	n/a	sí
Bicing Service	sí	n/a	sí	n/a	n/a	n/a	sí
Recommendations Service	sí	n/a	sí	n/a	n/a	n/a	sí
Routing Service	sí	n/a	sí	n/a	n/a	n/a	sí
Electric Charging Stations Service	sí	n/a	sí	n/a	n/a	n/a	sí

Taula 1: Taula indicadors integració BIG IoT

Tal i com es pot apreciar a la Taula 1, cada component desenvolupat compleix correctament amb els respectius indicadors, avalant així la correcta integració amb la resta de components de BIG IoT.

13 Planificació temporal i econòmica

13.1 Planificació general

El projecte BIG IoT va començar l'any 2016 i té una durada de tres anys. La implementació de la part del pilot de Barcelona d'aquest treball final de grau comença el Setembre de 2017 i acaba el mes de Juny de 2018.

A continuació s'exposa la planificació d'aquest treball final de grau; no és la planificació del pilot de Barcelona, ja que aquest és molt més extens i hi intervenen altres parts interessades.

La planificació s'ha estructurat en paquets de treball, cada paquet de treball conté tasques relacionades en temàtica o tecnologia. Els paquets de treball, com es pot observar al diagrama de Gantt, són consecutius. En canvi, les tasques incloses en aquests paquets, poden ser desenvolupades de manera concurrent.

13.2 Recursos

Per dur a terme aquest treball final de grau són necessaris recursos materials i de personal.

13.2.1 Personal

Per part de l'Inlab FIB, l'equip encarregat del pilot de Barcelona és format per:

- **Cap de projecte:** Persona encarregada de liderar l'equip.
- **Expert científic:** Persona que aporta coneixements específics per al pilot. A l'equip hi participen dues persones amb aquest perfil.
- **Desenvolupador sénior:** Persona amb experiència amb altres projectes, encarregada de liderar la presa de decisions a nivell tècnic i d'organització de l'equip de desenvolupament.
- **Desenvolupador junior:** Persona amb poca experiència, dedicada a desenvolupar el software necessari pel pilot.

Donat que aquest treball final de grau és una part del pilot de Barcelona, una part de l'equip del pilot ha participat activament en aquest treball: **Cap de projecte, desenvolupador sénior i junior.**

13.2.2 Material

Recurs	Tipus	Finalitat
Ordinador de sobretaula	Hardware	Per poder desenvolupar i escriure la memòria del projecte.
Servidor producció	Hardware	Per tal de desplegar els diferents artefactes que es generen durant el projecte.
IntelliJ IDEA	Eina de desenvolupament	Editor codi Java.
VSCode	Eina de desenvolupament	Editor de codi, utilitzat per desenvolupar web.
TeXstudio	Eina de desenvolupament	Editor Latex, per tal de realitzar la memòria.
Git, GitLab	Eina de desenvolupament	Control de versions i repositori per al codi.
Trello	Eina de gestió	Taulell virtual per a la gestió del projecte.
Slack	Eina de comunicació	Xat intern per tal de facilitar la comunicació entre l'equip.

Taula 2: Taula de recursos materials

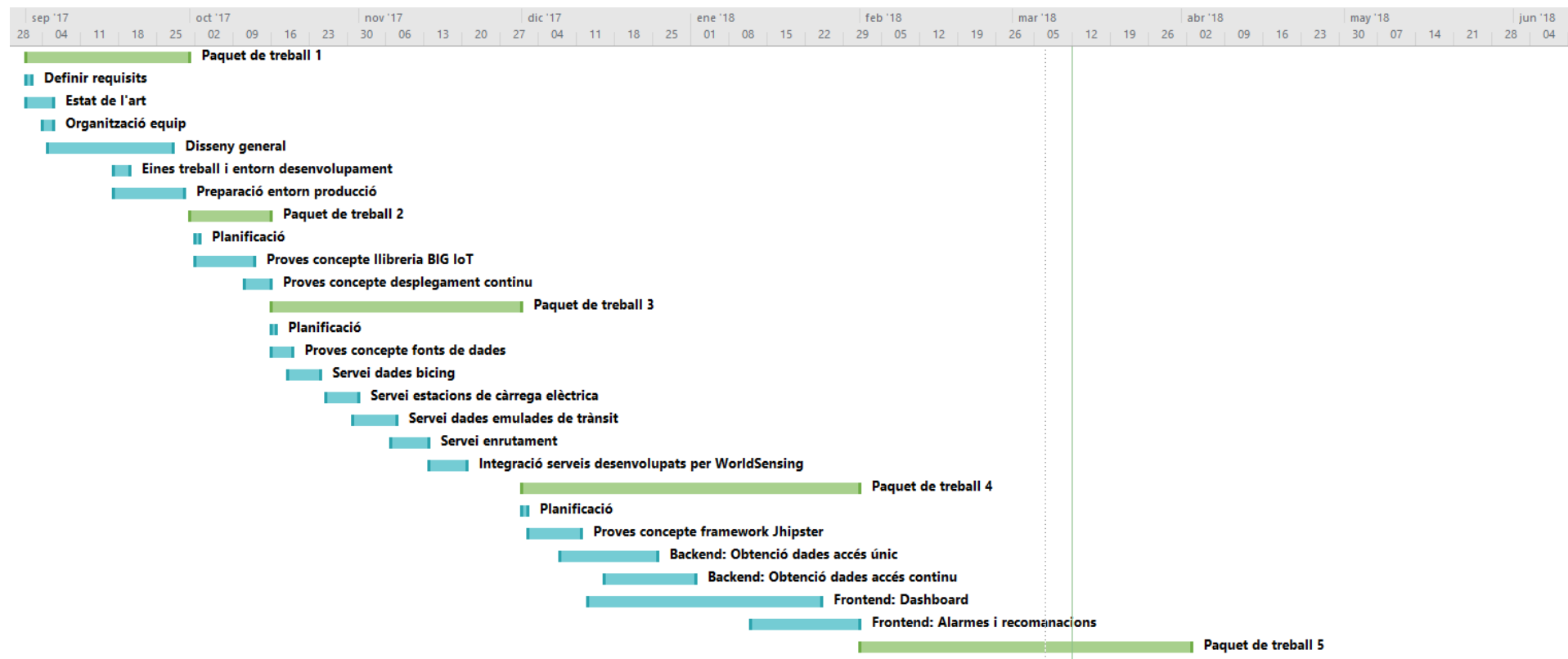


Figura 14: Diagrama de Gantt 1

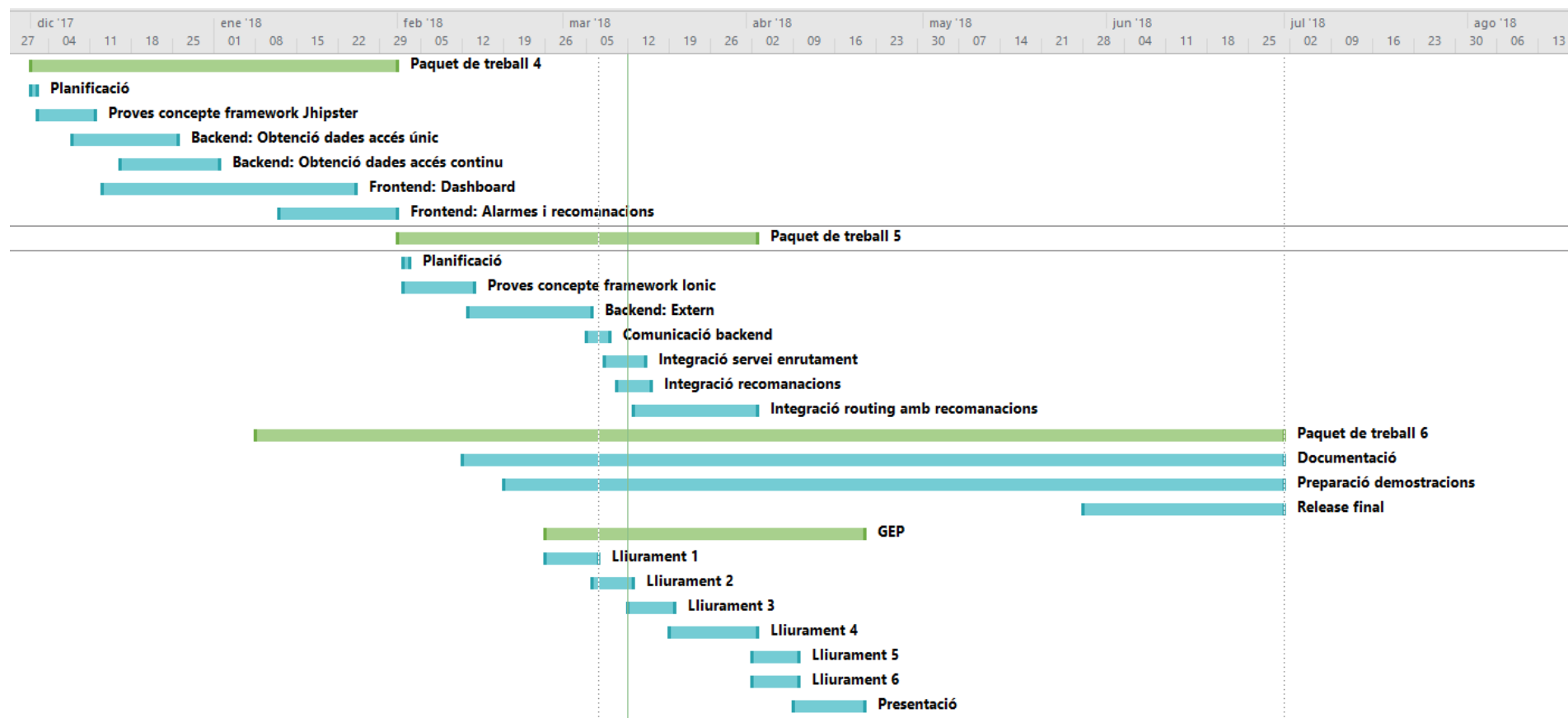


Figura 15: Diagrama de Gantt 2

13.3 Planificació econòmica

Tal i com mostra el diagrama de Gantt (Figures 14 i 15), aquest projecte té una durada de 41 setmanes, entre 01/09/2017-30/06/2018. No es computen dues setmanes festives. Aquest treball final de grau es desenvolupa a l'organització inlab FIB mitjançant un conveni amb la Universitat Politècnica de Catalunya.

$$(43 \text{ setmanes} - 2 \text{ setmanes festives}) * \frac{5 \text{ dies laborables}}{\text{setmana}} = 205 \text{ dies} \quad (1)$$

13.3.1 Recursos de personal

Tal i com s'ha mencionat anteriorment, la durada del projecte serà de 41 setmanes. La següent taula mostra les hores estimades que haurà de dedicar cada integrant de l'equip al llarg d'aquestes 41 setmanes. Es considera un participant per rol.

Rol	Preu/hora (€)	Hores estimades (h)	Cost (€)
Cap de projecte	35	80	2.800
Desenvolupador Sènior	20	60	1.200
Desenvolupador Junior	8	820	6.560
Expert científic	50	40	2.000
TOTAL		1000	12.560

Taula 3: Recursos personal

Recursos Hardware

S'utilitzen ordinadors de l'oficina, un per integrant de l'equip. Es suposen una mitjana de 750€ per dispositiu i una vida útil de 17.520 hores (2 anys). També es contracta un servidor al núvol a l'empresa ADW [11], amb un cost de 20€ al mes.

Recursos Software

Tot el software utilitzat al projecte és gratuït o bé amb llicència d'estudiant.

Concepte	Preu (€)	Vida útil (h)	Ús (h)	Cost (€)
Xiaomi Air 13'3	800	17.520	100	4,5
HP Compaq 8100	750	17.520	80	3,4
HP Compaq 8100	750	17.520	60	2,5
HP Compaq 8100	750	17.520	820	35,1
HP Compaq 8100	750	17.520	40	1,7
Servidor	20/mes		10 mesos	200
TOTAL				247,3

Taula 4: Recursos hardware

13.3.2 Costos indirectes

Tal i com s'ha mencionat anteriorment, aquest treball final de grau es desenvolupa al laboratori Inlab FIB. A continuació es consideren els costos indirectes que es deriven al desenvolupar un projecte de software a un espai físic de l'organització. Al concepte *Lloc de treball* ja es tenen en compte les despeses d'aigua, llum, gas, electricitat i material d'oficina.

Concepte	Ús	Preu (€/mes)	Duració (mesos)	Cost (€)
Connexió a internet	7%	70	10	49
Lloc de treball	10%	3.500	10	3.500
TOTAL				3.549

Taula 5: Recursos indirectes

13.3.3 Imprevistos

Al planificar aquest treball final de grau ja es consideren possibles desajustos temporals, de totes maneres, el projecte es podria allargar per tal de realitzar alguna tasca imprevista relacionada amb el pilot BIG IoT de Barcelona. Per exemple, en cas de que apareguin nous requisits per a les aplicacions i serveis desenvolupats, caldrà allargar el projecte fins que els requisits quedin satisfets. S'utilitzarien fons de la partida descrita a la secció següent.

També cal considerar el cost derivat de la fallada d'algun dels ordinadors del lloc de treball o servidor de producció. En cas de que es produeixi alguna averia als ordinadors de l'oficina, l'inlabFIB ja s'encarregaria de substituir el dispositiu avariats.

13.3.4 Contingències

Aquest treball final de grau forma part del conjunt de tasques que s'ha compromès l'inlabFIB al participar al projecte europeu BIG IoT. Una part del finançament total rebut és destinat a desenvolupar aquest treball. Es valora un 10% del pressupost total final de contingència per tal de fer front a possibles imprevistos.

Concepte	Percentatge	Preu	Cost
Recursos personal	10%	12.560	1.256
Recursos material	10%	247,3	24,7
Costos indirectes	10%	3.549	354,9
TOTAL			1.635,6

Taula 6: Taula contingències

13.3.5 Pressupost final

Finalment es sumen els costos associats al projecte.

Concepte	Cost (€)
Recursos personal	12.560
Recursos material	247,3
Costos indirectes	3.549
Contingències	1.635,6
TOTAL	17.992

Taula 7: Taula pressupost total

14 Anàlisi de la sostenibilitat

Seguidament s'analitza la sostenibilitat del treball final de grau mitjançant tres punts de vista diferents: ambiental, econòmic i social. Les següents reflexions comparteixen la hipòtesi d'un escenari a la ciutat de Barcelona on totes les plataformes, consumidores i proveïdors de dades, utilitzen les eines proporcionades per BIG IoT. En especial, s'utilitza intensament el *Marketplace*, on podem trobar totes les ofertes de dades dels diferents proveïdors de Barcelona.

L'anàlisi es realitza en base a la matriu de sostenibilitat proporcionada per l'assignatura de GEP.

14.1 Econòmica

El manteniment de l'ecosistema BIG IoT a la ciutat de Barcelona requeriria un *Marketplace* actiu constantment. També seria necessari un control i manteniment de les dades ofertes per part dels proveïdors. Podríem considerar que actualment les empreses ja realitzen controls semblants a les seves plataformes, per tant, no suposaria un cost extra.

La interoperabilitat i visibilitat de les dades que s'aconseguiria a l'escenari mencionat anteriorment permetria agilitzar la compra-venda de dades, incentivant així el mercat de l'IoT. La llibreria i el *Marketplace* permetrien als proveïdors de dades analitzar les característiques dels seus consumidors habituals, podent així millorar el model de negoci.

14.2 Ambiental

La implementació de les solucions que proposa el projecte BIG IoT a la ciutat de Barcelona implica que les plataformes proveïdors i consumidores de dades han d'afegir una nova capa al seu negoci. L'adició d'aquesta capa millora la interoperabilitat entre les diferents plataformes, reduint els costos de comunicació i transport de dades entre entitats. D'aquesta reducció se'n deriva una disminució del consum energètic, ja que es requereixen menys recursos al funcionar.

També caldria considerar el manteniment de l'ecosistema mencionat al punt anterior. Aquest manteniment requeriria destinar recursos energètics i econòmics.

14.3 Social

Suposant que la implementació del pilot BIG IoT de Barcelona fóra un èxit, es facilitaria l'accés a les dades per a qualssevol ciutadà, ja que les diferents ofertes de dades serien visibles per a tothom a través del *Marketplace*. Encara que aquest fet augmentaria la transparència de les dades facilitades per entitats públiques, les dades d'empreses privades podrien seguir distribuir-se sota llicència.

La millora a l'accés de dades també incentivaria el desenvolupament de noves aplicacions que en fessin ús.

15 Conclusions

Aquest apartat recull les valoracions, assoliment d'objectius, treball futur i contribucions d'aquest treball final de grau.

15.1 Assoliment d'objectius

L'objectiu principal d'aquest treball final de grau era desenvolupar aplicacions i serveis del pilot BIG IoT de Barcelona que facin ús de l'ecosistema i eines proporcionats pel projecte europeu. Al llarg d'aquesta memòria s'han exposat els diversos components desenvolupats que han fet ús de l'ecosistema BIG IoT per tal de comunicar-se i realitzar les pertinents regles de negoci. De les validacions i tests plantejats a l'apartat 12 se n'ha obtingut un balanç positiu, avalant així l'assoliment del principal objectiu d'aquest treball final de grau.

Pel que fa als objectius dels col·laboradors del pilot a Barcelona (Ajuntament, World-sensing i Seat) descrits a l'apartat 8.1, es poden classificar en dos grups: presència de les dades a l'entorn BIG IoT i visibilitat de les dades ofertes a les aplicacions desenvolupades. Els serveis BIG IoT que s'han desenvolupat, ofereixen les dades al *Marketplace* i asseguren l'assoliment dels objectius del primer grup. La visibilitat s'assegura mitjançant la *Traffic Information Center Tool* i la *BCN Traffic Info*. Els annexos d'aquesta memòria mostren captures de les aplicacions en qüestió.

Els objectius dels usuaris s'assoleixen mitjançant les funcionalitats que ofereixen l'aplicació web *Traffic Information Center Tool* i l'aplicació mòbil *BCN Traffic Info*. Als annexos es poden apreciar diverses funcionalitats descrites al llarg d'aquesta memòria.

Durant el desenvolupament d'aquest treball també han sorgit alguns dels possibles obstacles plantejats a l'apartat 5.1. Concretament, el *Marketplace* i la llibreria han estat actualitzats a noves versions que requerien canvis en el codi dels serveis i aplicacions existents. Encara que aquests canvis han alterat el correcte funcionament del *Marketplace* i han complicat el desenvolupament dels serveis i aplicacions, les metodologies àgils emprades ens han permès moderar l'impacte causat per aquest obstacle aconseguint assolir els objectius establerts.

15.2 Contribucions

En aquest treball final de grau s'han desenvolupat les aplicacions i serveis que conformen part de la prova pilot del projecte europeu BIG IoT a la ciutat de Barcelona.

D'acord amb el que s'ha explicat al llarg d'aquesta memòria, el projecte europeu proposa una solució a la problemàtica que presenta la falta d'interoperabilitat entre plataformes IoT. L'assoliment dels objectius d'aquest treball ha derivat en l'assoliment del propòsit principal del pilot BIG IoT a Barcelona: demostrar que les solucions proposades per BIG IoT podrien funcionar en un possible futur escenari en que BIG IoT tingués una forta presència a l'entorn IoT.

Per tant, la contribució principal d'aquest treball final de grau ha estat la realització de la prova de concepte i consegüent estudi de les solucions proposades pel projecte europeu BIG IoT. Concretament en l'aplicació d'aquestes solucions a la ciutat de Barcelona.

15.3 Treball futur

Les aplicacions i serveis desenvolupats han format part de la prova pilot de Barcelona en que es demostrava la viabilitat de les solucions proposades per BIG IoT. Aquests components desenvolupats poden ser millorats de dues maneres: millores encarades a generar un producte relacionat amb la mobilitat que pugui arribar a ser utilitzat per futurs clients, o bé, millores i addició de funcionalitats dels components desenvolupats dirigides a continuar estudiant la solució BIG IoT. És cert que moltes millores d'aquest últim tipus podrien incloure's a les del primer tipus.

Seguidament es detallen algunes de les possibles millores que podrien ser implementades en un futur:

- **Canvi proveïdor de l'algorisme de *routing*:** Per requisits provinents de BIG IoT, la plataforma que proporcionava les rutes i temps de viatge entre dos punts era proporcionada pels col·laboradors de Piedmont, Itàlia. Una de les principals àrees d'expertesa de l'inLab FIB és la gestió de la mobilitat utilitzant eines de simulació de trànsit. Per tant, l'aplicació d'aquests coneixements permetria modificar els algorismes d'enrutament i adaptar-los a les necessitats específiques de la mobilitat de la ciutat de Barcelona.
- **Noves alarmes:** El sistema d'alarmes s'ha desenvolupat de manera que sigui fàcil l'addició de noves dades a monitorar. Per tant, s'incentiva la possibilitat de monitorar noves dades que en un futur poden ser útils per a la gestió de la mobilitat de la ciutat.

- **Millores de producte:** Amb l'objectiu de convertir en un producte les aplicacions i serveis desenvolupats, caldria realitzar tests d'estrès del sistema i revisar els mètodes de desplegament.

15.4 Valoració personal

Realitzar aquest projecte a l'inLab FIB m'ha permès obtenir l'experiència de participar en un projecte europeu. Aquesta experiència ha estat especialment enriquidora pel fet de poder treballar amb col·laboradors d'arreu d'Europa, en que he apreciat les diverses formes d'encarar els diferents reptes que s'han presentat al llarg del projecte.

També ha sigut enriquidora l'experiència obtinguda al treballar amb l'equip de l'inLab FIB, format per companys amb perfils professionals diversos.

El desenvolupament d'aquest treball també ha representat un repte d'autoaprenentatge en tecnologies fins ara no coneixia, sempre amb el suport dels companys d'equip en situacions i problemes complicats.

Referències

- [1] Ayuntamiento de Barcelona. <http://ajuntament.barcelona.cat/es/>. [Online] [Consultat 03/03/2018].
- [2] BIG IoT – Bridging the Interoperability Gap of the Internet of Things. <http://big-iot.eu/>. (Consultat 02/20/2018).
- [3] FIWARE: The Open Source Platform For Our Smart Digital Future. <https://www.fiware.org/>. [Online] [Consultat 02/20/2018].
- [4] Git. <https://git-scm.com/>. [Online] [Consultat 03/01/2018].
- [5] Gitlab. <https://gitlab.com/>. [Online] [Consultat 03/01/2018].
- [6] Horizon 2020 - european commission - the eu framework programme for research and innovation. <https://ec.europa.eu/programmes/horizon2020/>. [Online] [Consultat 03/02/2018].
- [7] INTER-Iot - Interoperability Internet of Things. <http://www.inter-iot-project.eu/>. [Online] [Consultat 02/20/2018].
- [8] IoT-EPI. <http://iot-epi.eu/>. [Online] [Consultat 02/20/2018].
- [9] Open Data BCN — Servicio de datos abiertos del Ajuntament de Barcelona. <http://opendata-ajuntament.barcelona.cat/es>. [Online] [Consultat 02/26/2018].
- [10] SEAT España. <http://www.seat.es/>. [Online] [Consultat 03/03/2018].
- [11] Servidores Cloud SSD en España - ADW.es. <https://www.adw.es/servidores-cloud.html>. [Online] [Consultat 03/19/2018].
- [12] symbIoTe: Symbiosis of smart objects across IoT environments. <https://www.symbiote-h2020.eu/>.
- [13] W3C Web of Things at W3C. <https://www.w3.org/WoT/>. [Online] [Consultat 03/03/2018].
- [14] Worldsensing. <https://www.worldsensing.com/>. [Online] [Consultat 03/03/2018].
- [15] biotope project - building an iot open innovation ecosystem for connected smart objects. <http://www.biotope-project.eu/overview> <http://www.biotope-project.eu/>, 2017. [Online] [Consultat 02/26/2018].

-
- [16] Louis Columbus. 2017 Roundup Of Internet Of Things Forecasts. <https://www.forbes.com/sites/louiscolumbus/2017/12/10/2017-roundup-of-internet-of-things-forecasts>. [Online] [Consultat 02/15/2018].
 - [17] The Vicinity Consortium. Open virtual neighbourhood network to connect IoT infrastructures and smart objects. <http://vicinity2020.eu/vicinity/>, 2016. [Online] [Consultat 02/26/2018].
 - [18] Xavier Oriol, Ernest Teniente, and Guillem Rull. Tintin: a tool for incremental integrity checking of assertions in sql server. In *Advances in Database Technology-EDBT 2016, 19th International Conference on Extending Database Technology, Bordeaux, France, March 15-16, Proceedings*, pages 632–635, 2016.

A Captures de pantalla del *Traffic Information Center Tool*

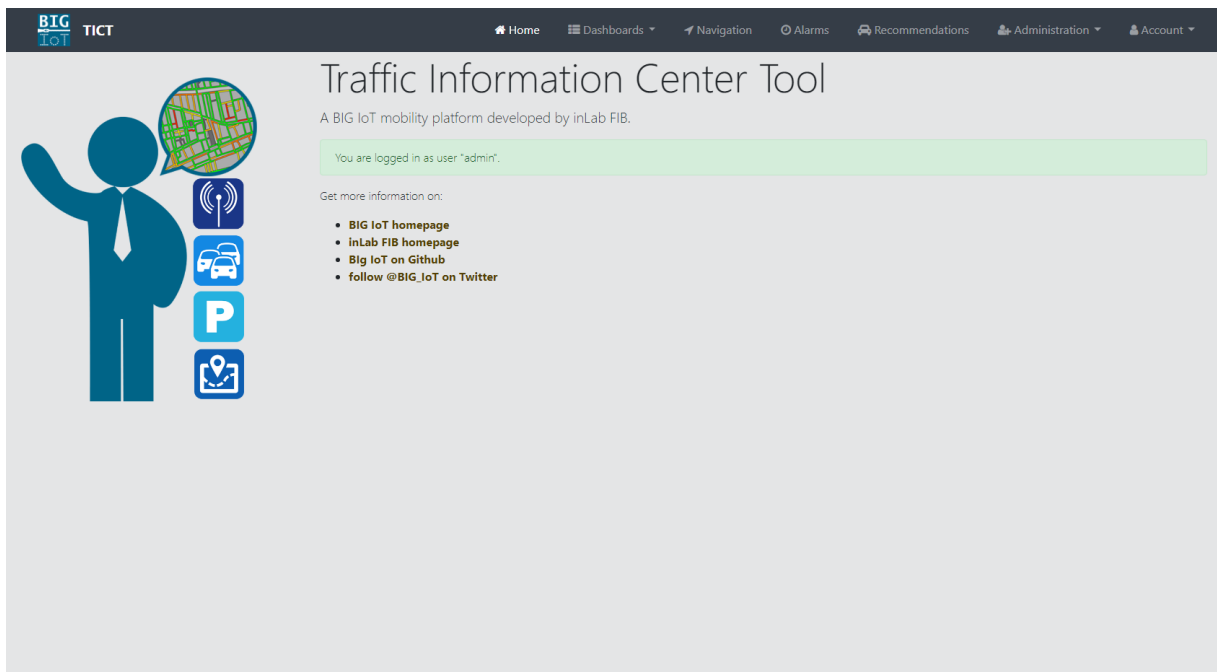


Figura 16: Pàgina de benvinguda

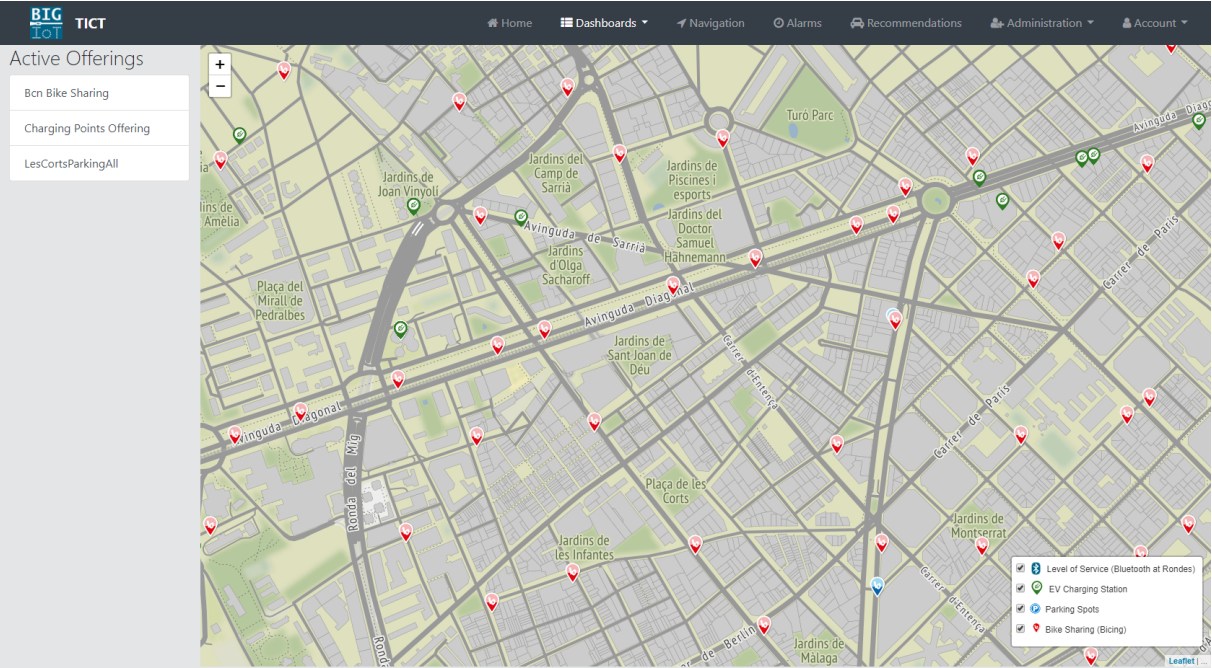


Figura 17: Vista del mapa amb dades de mobilitat 1

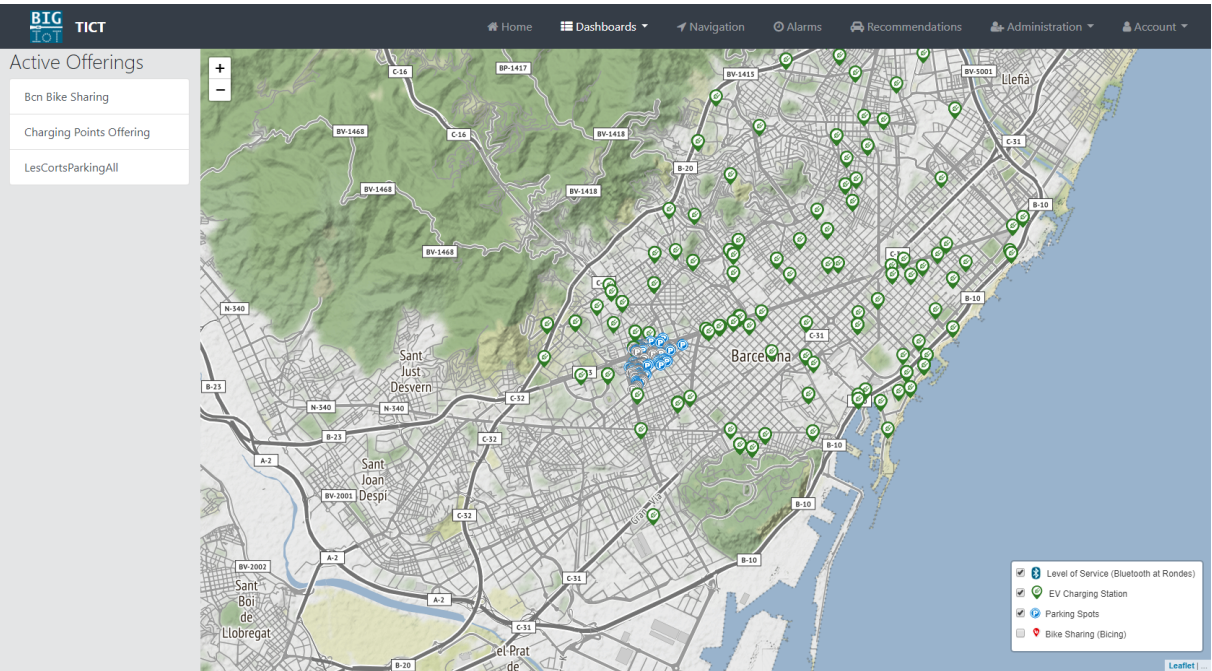


Figura 18: Vista del mapa amb dades de mobilitat 2

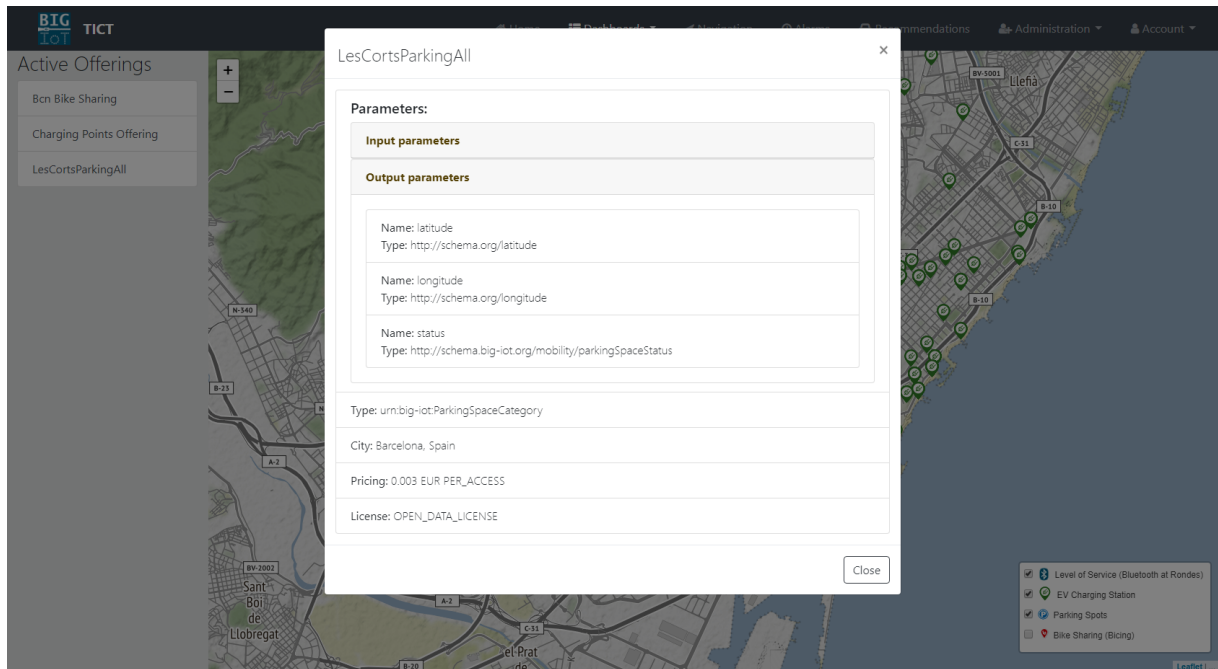


Figura 19: Vista informació de l'oferta de dades

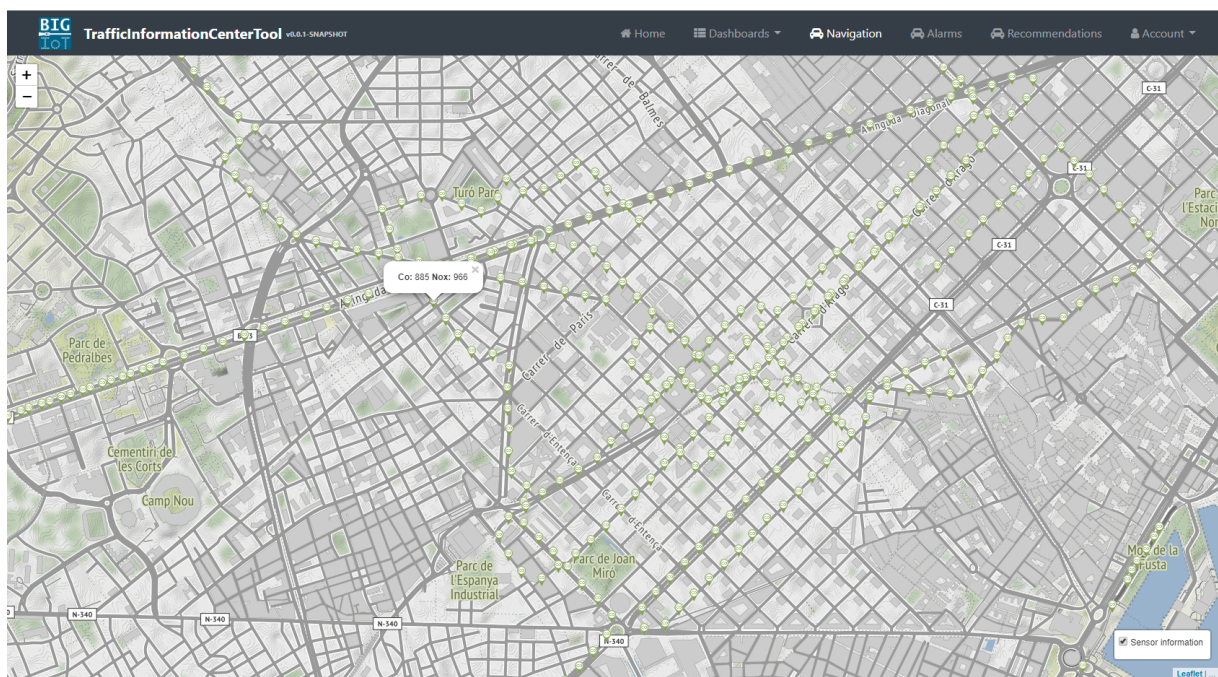


Figura 20: Vista del mapa amb dades de contaminació

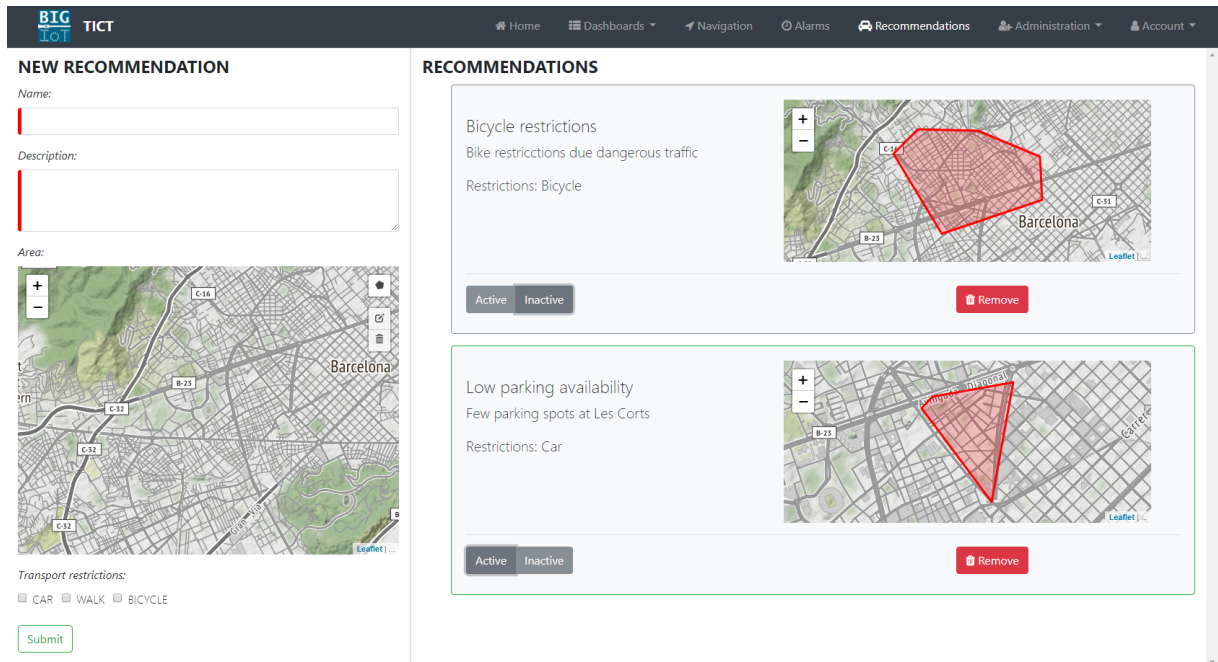


Figura 21: Vista gestió recomanacions

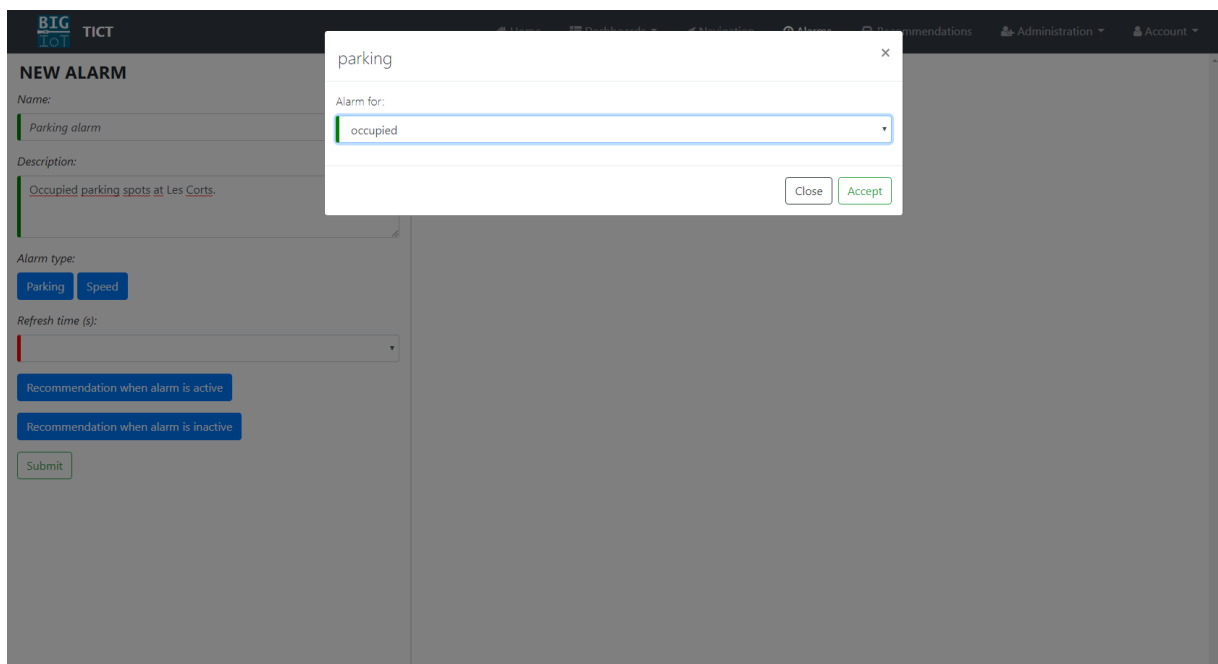


Figura 22: Vista creació alarmes (opció aparcament)

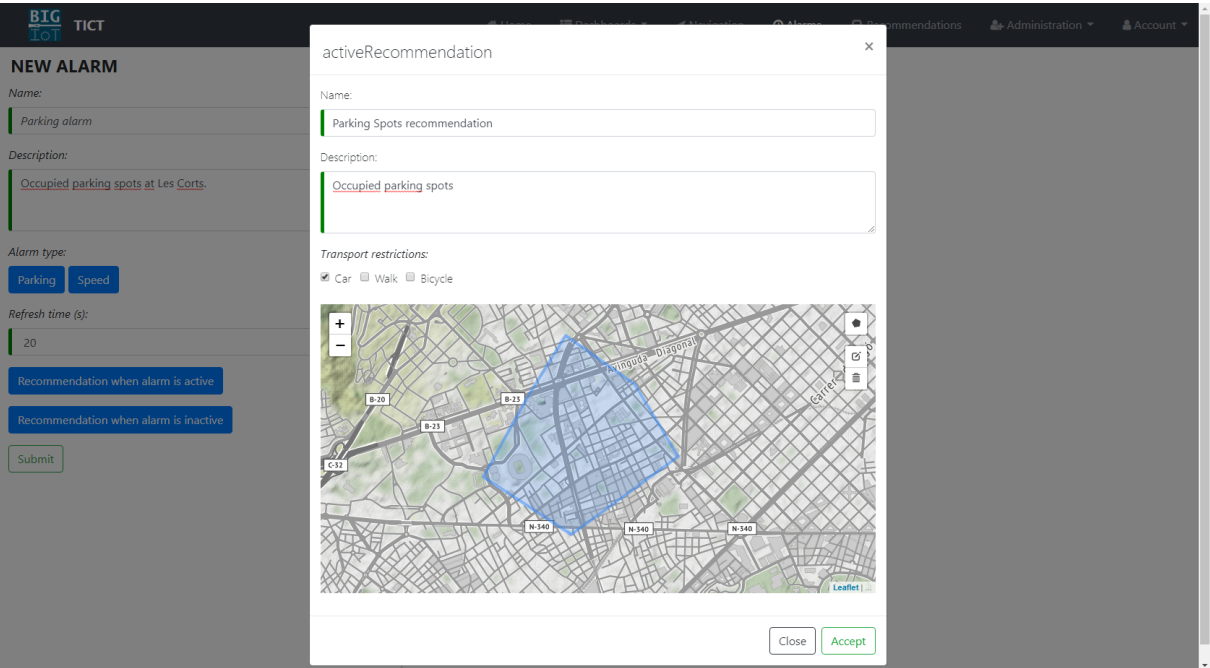


Figura 23: Vista creació alarmes (recomanació associada)

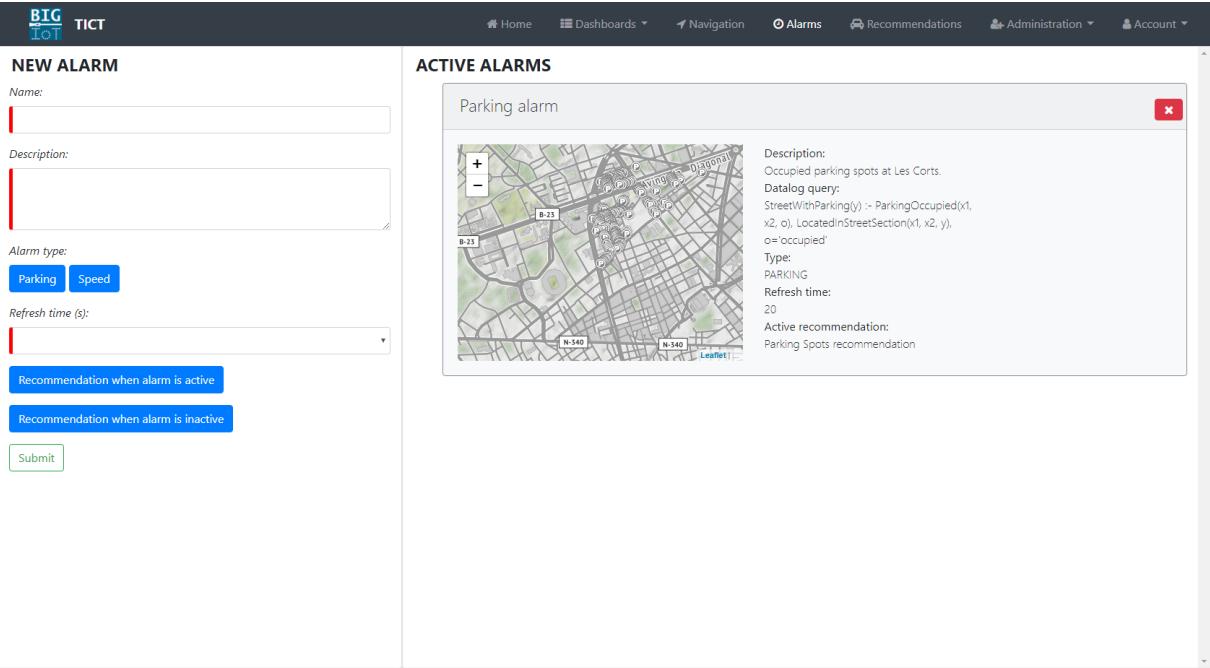


Figura 24: Vista gestió alarmes

B Captures de pantalla de la *BCN Traffic Info*

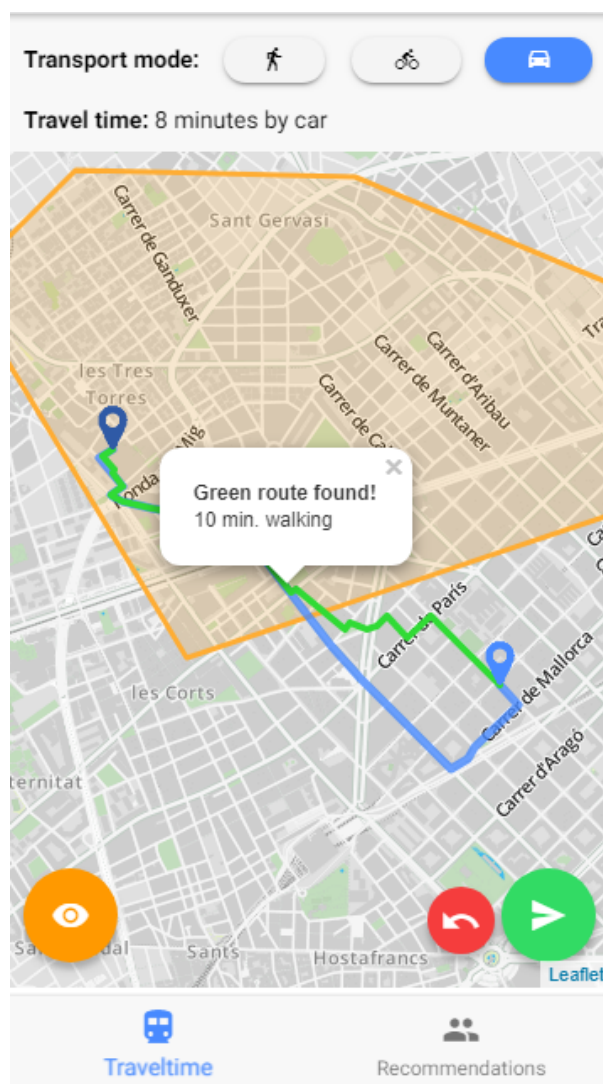


Figura 25: Vista ruta i temps de viatge amb recomanacions actives

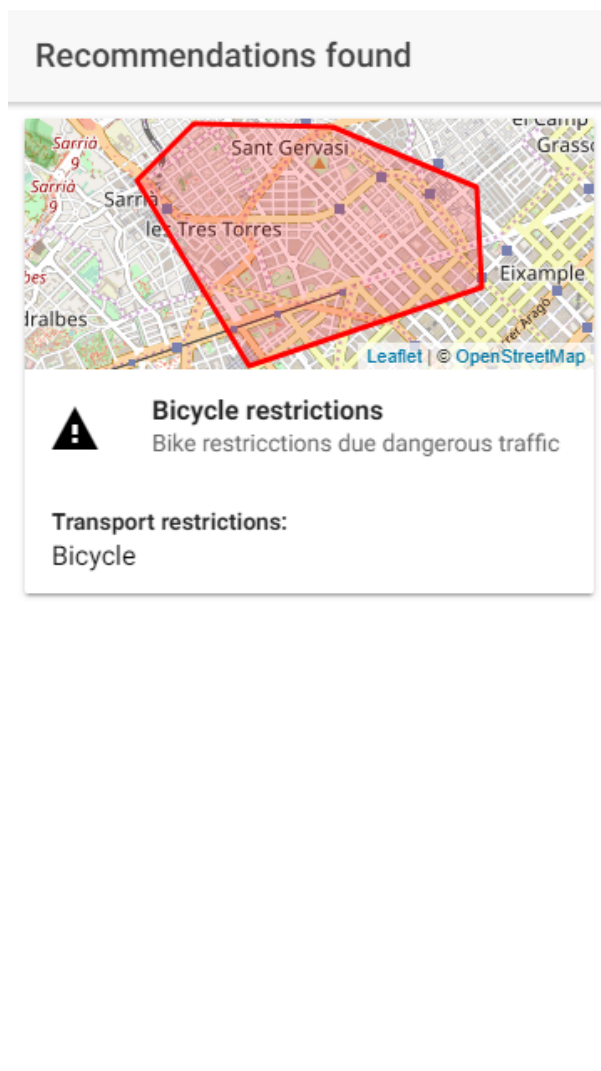


Figura 26: Vista recomanacions en ruta

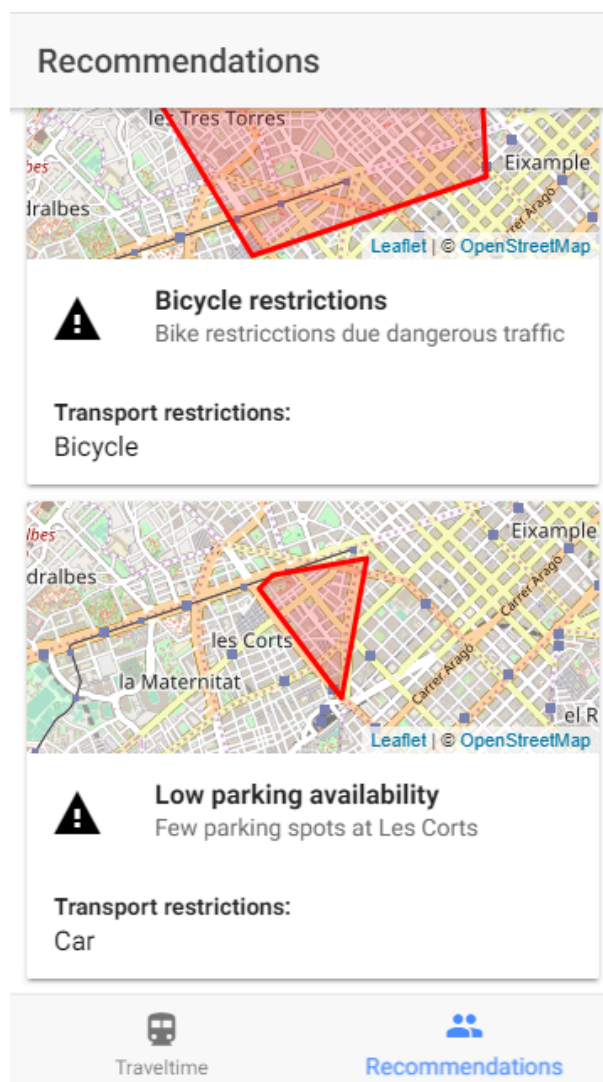


Figura 27: Vista de totes les recomanacions actives a la ciutat